Dépannage du cluster FTD (Firepower Threat Defense)

Table des matières

- Introduction
- Conditions préalables
  - Exigences
  - Composants utilisés
- Informations générales
- Configurer
- Notions de base des clusters
  - Architecture NGFW
  - Captures de cluster
  - Activation des captures de cluster
    - Captures FXOS
    - Captures du plan de données
    - Capturer les traces
    - Capture CCL
    - Réinjecter Masquer
    - Abandons ASP
    - Effacer une capture
    - Arrêter une capture
    - Collecter une capture
    - Supprimer une capture
    - Flux déchargés
  - Messages CCL (Cluster Control Link)
  - Messages CCP (Cluster Control Point)
  - Mécanisme de vérification de l'état de santé des clusters (HC)
  - Scénarios de défaillance HC de cluster
    - Topologie
    - Scénario 1 - CCLosion de communication pendant ~4 secondes ou plus dans les deux sens
    - Analyse
    - Restauration des communications CCL
    - Scénario 2 - CCLorte de communication pendant ~3-4 secondes dans les deux sens
    - Analyse
    - Scénario 3 - CCLorte de communication pendant ~3-4 secondes dans les deux sens
    - Analyse
    - Scénario 4 : perte de communication CCL d'environ 3 à 4 secondes
    - Analyse
    - Scénario 5
  - Établissement de la connexion du plan de données du cluster
    - Points de capture NGFW
  - Notions de base sur les rôles de flux des unités
  - Études de cas sur Cluster Connection Establishment
    - Topologie

- Étude de cas 1. Trafic symétrique (le propriétaire est également le directeur)
- Étude de cas 2. Trafic symétrique (propriétaire différent du directeur)
- Étude de cas 3. Trafic asymétrique (le directeur achemine le trafic)
- Unit-1-1 (propriétaire)
- Étude de cas 4. Trafic asymétrique (le propriétaire est le directeur)
- Étude de cas 5. Trafic asymétrique (le propriétaire est différent du directeur)
- Étude de cas 6. Trafic asymétrique (en ligne, le propriétaire est le directeur)
- Trafic de retour (TCP SYN/ACK)
- Étude de cas 7. Trafic asymétrique (en ligne, le propriétaire est différent du directeur)
- Transfert du trafic (TCP SYN)
- Trafic de retour (TCP SYN/ACK)

Dépannage

- Présentation du dépannage de cluster
- Problèmes de plan de données de cluster
  - Problèmes courants de NAT/PAT
  - Utilisation élevée de la plage du pool PAT en raison du trafic provenant de ports faibles qui provoque un déséquilibre IPI du cluster
  - Les connexions soumises à la PAT échouent
  - Améliorations PAT de mise en grappe ASA et FTD (post-9.15 et 6.7)
- Informations de dépannage supplémentaires sur la PAT
  - Syslogs du plan de données FTD (post-6.7/9.15.1)
  - Commandes show
- Foire aux questions (FAQ) pour la PAT post-6.7/9.15.1
- Traitement des fragments
  - Problèmes ACI

Problèmes de plan de contrôle de cluster

- L'unité ne peut pas joindre le cluster
  - Taille MTU sur CCL
  - Non-Concordance D'Interface Entre Les Unités De Cluster
- Problème d'interface Data/Port-Channel
  - Split-brain dû à des problèmes d'accessibilité sur la CCL
  - Cluster désactivé en raison d'interfaces Port Channel de données suspendues
- Problèmes de stabilité des clusters
  - Suivi FXOS
  - Disque plein
  - Protection Contre Les Débordements
  - Mode simplifié

Informations connexes

Introduction

Ce document décrit le dépannage d'une configuration de cluster sur le pare-feu de nouvelle génération Firepower (NGFW).

Conditions préalables

Exigences

Cisco recommande que vous ayez connaissance de ces sujets (voir la section Informations connexes pour les liens) :

- Architecture de plate-forme Firepower
- Configuration et fonctionnement du cluster Firepower
- Familiarité avec l'interface de ligne de commande FTD et Firepower eXtensible Operating System (FXOS)
- Logiciels NGFW/journaux du plan de données
- NGFW/traceur de paquets de plan de données
- Captures FXOS/plan de données

Composants utilisés

- MATÉRIEL : Firepower 4125
- Logiciel : 6.7.0 (Build 65) - plan de données 9.15(1)

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l’incidence possible des commandes.

Informations générales

La plupart des éléments abordés dans ce document s'appliquent également au dépannage de cluster ASA (Adaptive Security Appliance).

Configurer

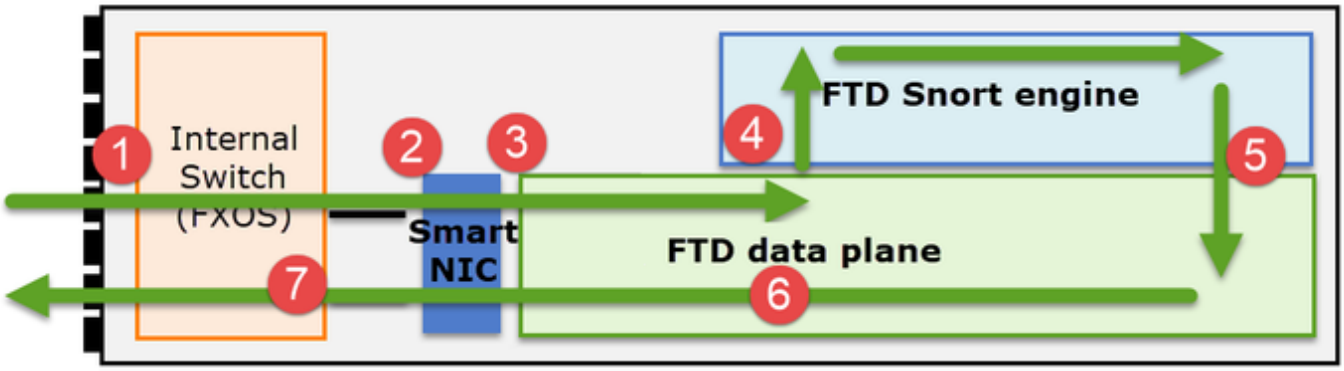
La partie configuration d'un déploiement de cluster est traitée dans les guides de configuration FMC et FXOS :

- Mise en grappe pour Firepower Threat Defense
- Déploiement d'un cluster pour Firepower Threat Defense pour une évolutivité et une haute disponibilité

Notions de base des clusters

Architecture NGFW

Il est important de comprendre comment une gamme Firepower 41xx ou 93xx gère les paquets de transit :



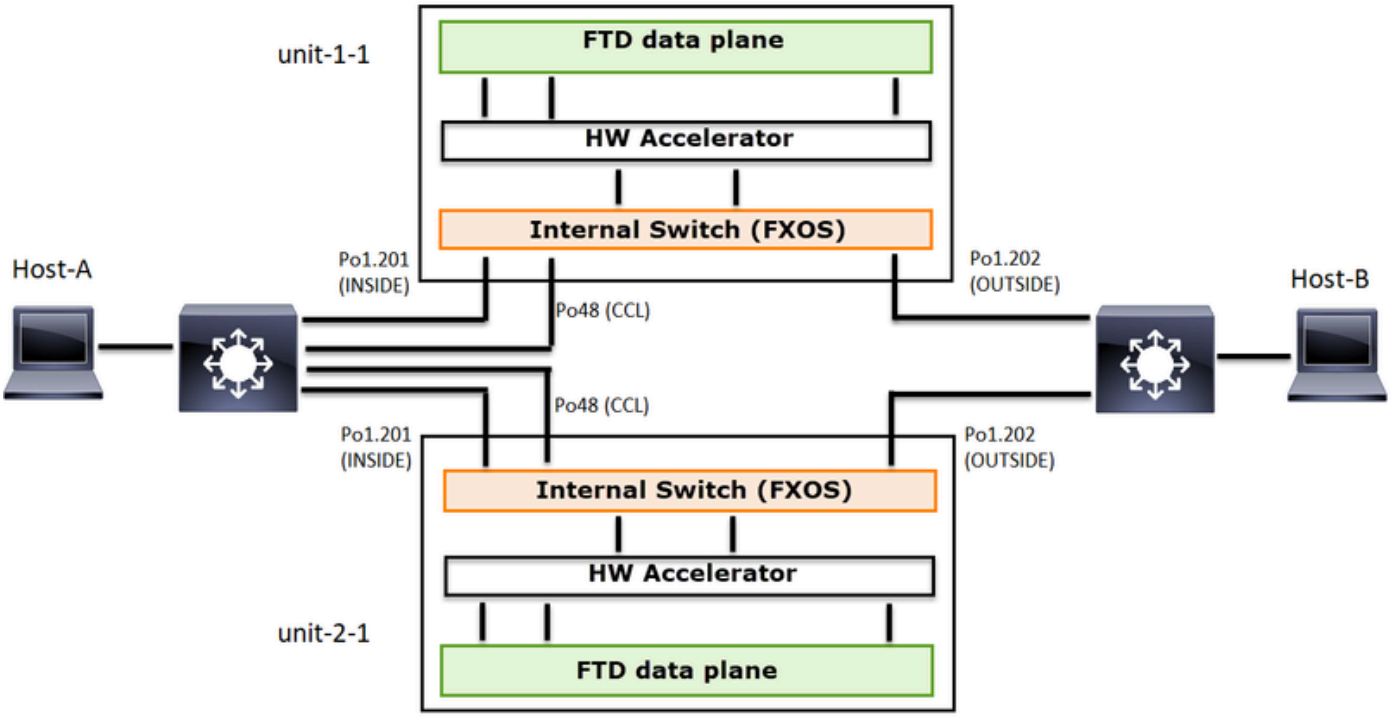
1. Un paquet entre dans l'interface d'entrée et est géré par le commutateur interne du châssis.
2. Le paquet passe par la carte réseau intelligente. Si le flux est déchargé (accélération matérielle), le paquet est traité uniquement par la carte réseau intelligente, puis renvoyé au réseau.
3. Si le paquet n'est pas déchargé, il entre dans le plan de données FTD qui effectue principalement des vérifications L3/L4.
4. Si la politique l'exige, le paquet est inspecté par le moteur Snort (principalement inspection L7).
5. Le moteur Snort renvoie un verdict (par exemple, autoriser ou bloquer) pour le paquet.
6. Le plan de données abandonne ou transfère le paquet en fonction du verdict de Snort.
7. Le paquet sort du châssis par le commutateur interne du châssis.

Captures de cluster

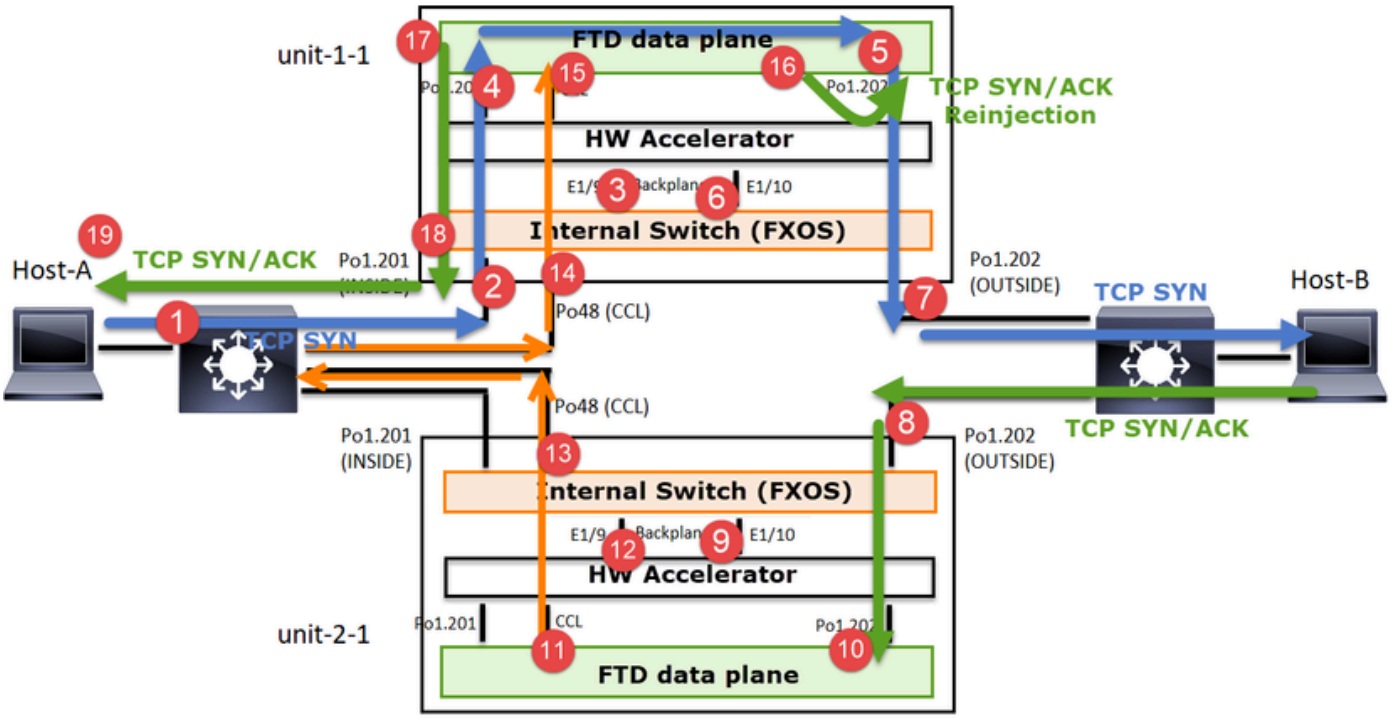
Les appliances Firepower fournissent plusieurs points de capture qui offrent une visibilité sur les flux de transit. Lorsque vous dépannez et activez les captures de cluster, les principaux défis sont les suivants :

- Le nombre de captures augmente en même temps que le nombre d'unités du cluster.
- Vous devez connaître la façon dont le cluster gère un flux spécifique pour pouvoir suivre le paquet à travers le cluster.

Ce schéma illustre un cluster de 2 unités (par exemple, FP941xx/FP9300) :



Dans le cas d'un établissement de connexion TCP asymétrique, un échange SYN, SYN/ACK TCP ressemble à ceci :



Trafic de transfert :

1. TCP SYN est envoyé de l'hôte A à l'hôte B.
2. TCP SYN arrive sur le châssis (un des membres de Po1).
3. TCP SYN est envoyé au plan de données via l'une des interfaces de fond de panier du châssis (par exemple, E1/9, E1/10, etc.).
4. TCP SYN arrive sur l'interface d'entrée du plan de données (Po1.201/INSIDE). Dans cet exemple, unit1-1 prend possession du flux, effectue la randomisation ISN (Initial Sequence

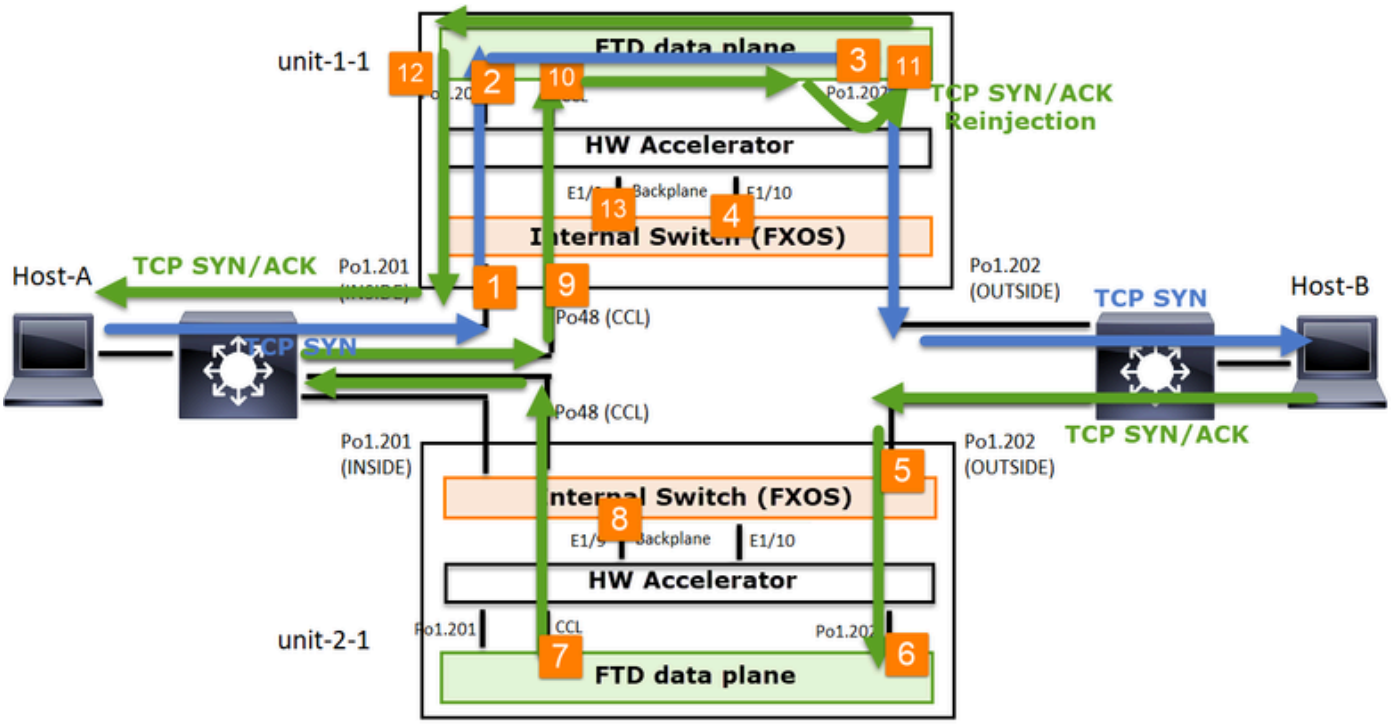
Number) et code les informations de propriété (cookie) dans le numéro de séquence.
5. TCP SYN est envoyé depuis Po1.202/OUTSIDE (interface de sortie du plan de données).
6. TCP SYN arrive sur l'une des interfaces de fond de panier du châssis (par exemple, E1/9, E1/10, etc.).
7. TCP SYN est envoyé de l'interface physique du châssis (l'un des membres de Po1) vers l'hôte B.

Trafic de retour :

8. TCP SYN/ACK est envoyé depuis l'hôte B et arrive sur l'unité 2-1 (l'un des membres de Po1).
9. TCP SYN/ACK est envoyé au plan de données via l'une des interfaces de fond de panier du châssis (par exemple, E1/9, E1/10, etc.).
10. TCP SYN/ACK arrive sur l'interface d'entrée du plan de données (Po1.202/OUTSIDE).
11. TCP SYN/ACK est envoyé à partir de la liaison de contrôle de cluster (CCL) vers l'unité 1-1. Par défaut, ISN est activé. Ainsi, le redirecteur trouve les informations de propriétaire pour les SYN+ACK TCP sans l'implication du directeur. Pour les autres paquets ou lorsque le RNIS est désactivé, le directeur est interrogé.
12. TCP SYN/ACK arrive sur l'une des interfaces de fond de panier du châssis (par exemple, E1/9, E1/10, etc.).
13. TCP SYN/ACK est envoyé de l'interface physique du châssis (l'un des membres de Po48) vers l'unité 1-1.
14. TCP SYN/ACK arrive sur l'unité 1-1 (l'un des membres de Po48).
15. TCP SYN/ACK est transmis via l'une des interfaces de fond de panier du châssis à l'interface port-channel CCL du plan de données (cluster de noms).
16. Le plan de données réinjecte le paquet TCP SYN/ACK dans l'interface de plan de données Po1.202/OUTSIDE.
17. TCP SYN/ACK est envoyé de Po1.201/INSIDE (interface de sortie du plan de données) vers l'hôte A.
18. Le protocole TCP SYN/ACK traverse l'une des interfaces de fond de panier du châssis (par exemple, E1/9, E1/10, etc.) et sort de l'un des membres de Po1.
19. TCP SYN/ACK arrive sur l'hôte A.

Pour plus d'informations sur ce scénario, consultez la section correspondante des Études de cas d'établissement d'une connexion de cluster.

Sur la base de cet échange de paquets, tous les points de capture de cluster possibles sont les suivants :



Pour la capture du trafic de transfert (par exemple, TCP SYN) sur :

1. Interface physique du châssis (par exemple, membres Po1). Cette capture est configurée à partir de l'interface utilisateur de Chassis Manager (CM) ou de l'interface de ligne de commande de CM.
2. Interface d'entrée du plan de données (par exemple, Po1.201 INSIDE).
3. Interface de sortie du plan de données (par exemple, Po1.202 OUTSIDE).
4. Interfaces de fond de panier du châssis. Le FP4100 comporte 2 interfaces de fond de panier. Sur le FP9300, il y en a un total de 6 (2 par module). Puisque vous ne savez pas dans quelle interface le paquet arrive, vous devez activer la capture sur toutes les interfaces.

Pour la capture du trafic de retour (par exemple, TCP SYN/ACK) sur :

5. Interface physique du châssis (par exemple, membres Po1). Cette capture est configurée à partir de l'interface utilisateur de Chassis Manager (CM) ou de l'interface de ligne de commande de CM.
6. Interface d'entrée du plan de données (par exemple, Po1.202 OUTSIDE).
7. Comme le paquet est redirigé, le point de capture suivant est le plan de données CCL.
8. Interfaces de fond de panier du châssis. Là encore, vous devez activer la capture sur les deux interfaces.
9. Interfaces membres CCL du châssis Unit-1-1.
10. Interface CCL du plan de données (cluster de nom).
11. Interface d'entrée (Po1.202 OUTSIDE). Il s’agit du paquet réinjecté de CCL vers le plan de données.
12. Interface de sortie du plan de données (par exemple, Po1.201 INSIDE).
13. Interfaces de fond de panier du châssis.

Activation des captures de cluster

Captures FXOS

Le processus est décrit dans le Guide de configuration de FXOS : Capture de paquets

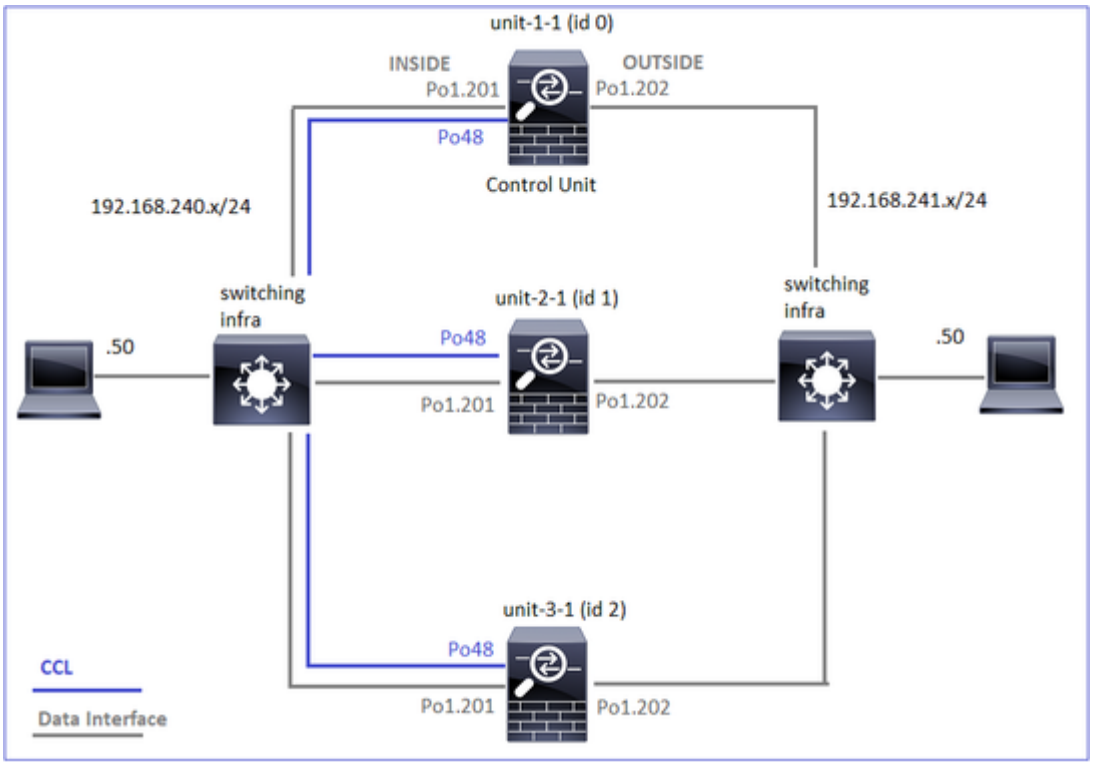
---

Remarque : Les captures FXOS ne peuvent être prises que dans la direction d'entrée du point de vue du commutateur interne.

Captures du plan de données

La méthode recommandée pour activer la capture sur tous les membres du cluster est avec la commande cluster exec.

Prenons l'exemple d'un cluster à 3 unités :



Pour vérifier s'il existe des captures actives dans toutes les unités de cluster, utilisez cette commande :

<#root>

```
firepower#
```

`cluster exec show capture`

```
unit-1-1(LOCAL):*********************************************************

unit-2-1:****************************************************************

unit-3-1:****************************************************************
firepower#
```

Pour activer une capture de plan de données sur toutes les unités sur Po1.201 (INSIDE) :

<#root>

```
firepower#
```

```
cluster exec capture CAPI interface INSIDE
```

Il est fortement recommandé de spécifier un filtre de capture et, si vous prévoyez un trafic important, d'augmenter la mémoire tampon de capture :

<#root>

```
firepower#
```

```
cluster exec capture CAPI buffer 33554432 interface INSIDE match tcp host 192.168.240.50 host 192.168.24
```

Vérification :

<#root>

```
firepower#
```

```
cluster exec show capture
```

```
unit-1-1(LOCAL):**********************************************************
capture CAPI type raw-data buffer 33554432 interface INSIDE [Capturing - 5140 bytes]
  match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-2-1:**********************************************************
capture CAPI type raw-data buffer 33554432 interface INSIDE [Capturing - 260 bytes]
  match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:**********************************************************
capture CAPI type raw-data buffer 33554432 interface INSIDE [Capturing - 0 bytes]
  match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

Pour afficher le contenu de toutes les captures (ce résultat peut être très long) :

<#root>

```
firepower#
```

```
terminal pager 24
```

```
firepower#
```

```
cluster exec show capture CAPI


unit-1-1(LOCAL):*******************************************************
21 packets captured

1: 11:33:09.879226 802.1Q vlan#201 P0 192.168.240.50.45456 > 192.168.241.50.80: S 2225395909:2225395909
2: 11:33:09.880401 802.1Q vlan#201 P0 192.168.241.50.80 > 192.168.240.50.45456: S 719653963:719653963(0
3: 11:33:09.880691 802.1Q vlan#201 P0 192.168.240.50.45456 > 192.168.241.50.80: . ack 719653964 win 229
4: 11:33:09.880783 802.1Q vlan#201 P0 192.168.240.50.45456 > 192.168.241.50.80: P 2225395910:2225396054

unit-2-1:*************************************************************
0 packet captured
0 packet shown

unit-3-1:*************************************************************
0 packet captured
0 packet shown
```

Capturer les traces

Si vous voulez voir comment les paquets entrants sont traités par le plan de données sur chaque unité, utilisez le mot clé trace. Il suit les 50 premiers paquets entrants. Vous pouvez tracer jusqu'à 1 000 paquets entrants.



Remarque : Si plusieurs captures sont appliquées à une interface, vous ne pouvez suivre qu'une seule fois un seul paquet.

Pour suivre les 1 000 premiers paquets entrants sur l'interface OUTSIDE sur toutes les unités de cluster :

```
<#root>

firepower#

cluster exec cap CAPO int OUTSIDE buff 33554432 trace trace-count 1000 match tcp host 192.168.240.50 hos
```

Une fois que vous avez capturé le flux d'intérêt, vous devez vous assurer que vous suivez les paquets d'intérêt sur chaque unité. Il est important de se rappeler qu'un paquet spécifique peut être #1 sur l'unité 1-1, mais #2 sur une autre unité, etc.

Dans cet exemple, vous pouvez voir que le paquet SYN/ACK est le paquet #2 sur l'unité-2-1, mais le paquet #1 sur l'unité-3-1 :

```
<#root>

firepower#
```

```
cluster exec show capture CAPO | include S.*ack


unit-1-1(LOCAL):*******************************************************
1: 12:58:31.117700 802.1Q vlan#202 P0 192.168.240.50.45468 > 192.168.241.50.80: S 441626016:441626016(0)
2: 12:58:31.118341 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45468:
S
 301658077:301658077(0)
ack
 441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>

unit-2-1:*******************************************************

unit-3-1:*******************************************************
1: 12:58:31.111429 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45468:
S
 301658077:301658077(0)
ack
 441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>
```

Pour suivre le paquet #2 (SYN/ACK) sur l'unité locale :

```
<#root>
firepower#
cluster exec show cap CAPO packet-number 2 trace


unit-1-1(LOCAL):*******************************************************
2: 12:58:31.118341 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45468:
S
 301658077:301658077(0)
ack
 441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>
Phase: 1
Type: CAPTURE
Subtype:
Result: ALLOW
Config:
Additional Information:
MAC Access list
...
```

Pour suivre le même paquet (SYN/ACK) sur l'unité distante :

<#root>

```
firepower#
```

cluster exec unit unit-3-1 show cap CAPO packet-number 1 trace

```
1: 12:58:31.111429 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45468:
```

S

```
 301658077:301658077(0)
```

ack

```
 441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>
Phase: 1
Type: CAPTURE
Subtype:
Result: ALLOW
Config:
Additional Information:
MAC Access list
...
```

Capture CCL

Pour activer la capture sur la liaison CCL (sur toutes les unités) :

<#root>

```
firepower#
```

cluster exec capture CCL interface cluster

```
unit-1-1(LOCAL):*******************************************************

unit-2-1:***************************************************************

unit-3-1:***************************************************************
```

Réinjecter Masquer

Par défaut, une capture activée sur une interface de données de plan de données affiche tous les paquets :

- Ceux qui arrivent du réseau physique
- Ceux qui sont réinjectés à partir de la CCL

Si vous ne voulez pas voir les paquets réinjectés, utilisez l'option reinject-hide. Cela peut être utile si vous voulez vérifier si un flux est asymétrique :

```
<#root>

firepower#

cluster exec capture CAPI_RH reinject-hide interface INSIDE match tcp host 192.168.240.50 host 192.168.2
```

Cette capture vous montre uniquement ce que l'unité locale reçoit réellement sur l'interface spécifique directement du réseau physique, et non des autres unités de cluster.

Abandons ASP

Si vous voulez vérifier les pertes logicielles pour un flux spécifique, vous pouvez activer la capture asp-drop. Si vous ne savez pas sur quelle raison vous concentrer, utilisez le mot clé all. En outre, si vous n'êtes pas intéressé par la charge utile du paquet, vous pouvez spécifier le mot clé headers-only. Cela vous permet de capturer 20 à 30 fois plus de paquets :

```
<#root>

firepower#

cluster exec cap ASP type asp-drop all buffer 33554432 headers-only


unit-1-1(LOCAL):*********************************************************

unit-2-1:************************************************************

unit-3-1:************************************************************
```

En outre, vous pouvez spécifier les IP qui vous intéressent dans la capture ASP :

```
<#root>

firepower#

cluster exec cap ASP type asp-drop all buffer 33554432 headers-only


match ip host 192.0.2.100 any
```

Effacer une capture

Pour effacer la mémoire tampon de toute capture exécutée dans toutes les unités de cluster. Cela n'arrête pas les captures, mais efface uniquement les tampons :

```
<#root>

firepower#
```

```
cluster exec clear capture /all

unit-1-1(LOCAL):**********************************************************
unit-2-1:**********************************************************
unit-3-1:**********************************************************
```

Arrêter une capture

Il existe deux façons d'arrêter une capture active sur toutes les unités de cluster. Plus tard, vous pourrez reprendre.

Voie 1 :

```
<#root>
firepower#
cluster exec cap CAPI stop

unit-1-1(LOCAL):**********************************************************
unit-2-1:**********************************************************
unit-3-1:**********************************************************
```

Pour reprendre :

```
<#root>
firepower#
cluster exec no capture CAPI stop

unit-1-1(LOCAL):**********************************************************
unit-2-1:**********************************************************
unit-3-1:**********************************************************
```

Voie 2 :

```
<#root>
firepower#
cluster exec no capture CAPI interface INSIDE
```

```
unit-1-1(LOCAL):*******************************************************

unit-2-1:*******************************************************

unit-3-1:*******************************************************
```

Pour reprendre :

```
<#root>

firepower#

cluster exec capture CAPI interface INSIDE

unit-1-1(LOCAL):*******************************************************

unit-2-1:*******************************************************

unit-3-1:*******************************************************
```

Collecter une capture

Il existe plusieurs façons d'exporter une capture.

Voie 1 - Vers un serveur distant :

Cela vous permet de télécharger une capture depuis le plan de données vers un serveur distant (par exemple, TFTP). Les noms de capture sont automatiquement modifiés pour refléter l'unité source :

```
<#root>

firepower#

cluster exec copy /pcap capture:CAPI tftp://192.168.240.55/CAPI.pcap

unit-1-1(LOCAL):*******************************************************

Source capture name [CAPI]?

Address or name of remote host [192.168.240.55]?

Destination filename [CAPI.pcap]?

INFO: Destination filename is changed to unit-1-1_CAPI.pcap !!!!!!!

81 packets copied in 0.40 secs

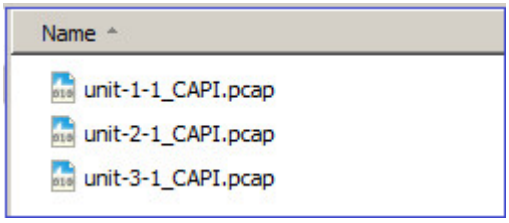
unit-2-1:*******************************************************
```

```
INFO: Destination filename is changed to unit-2-1_CAPI.pcap !


unit-3-1:***************************************************************

INFO: Destination filename is changed to unit-3-1_CAPI.pcap !
```

Les fichiers pcap téléchargés :



Méthode 2 - Récupérer les captures du FMC :

Cette méthode s'applique uniquement au FTD. Tout d'abord, copiez la capture sur le disque FTD :

<#root>

```
firepower#

cluster exec copy /pcap capture:CAPI disk0:CAPI.pcap


unit-1-1(LOCAL):*********************************************************

Source capture name [CAPI]?

Destination filename [CAPI.pcap]?
!!!!!
62 packets copied in 0.0 secs
```

À partir du mode expert, copiez le fichier du répertoire /mnt/disk0/ vers le répertoire /ngfw/var/common/ :

<#root>

```
>

expert

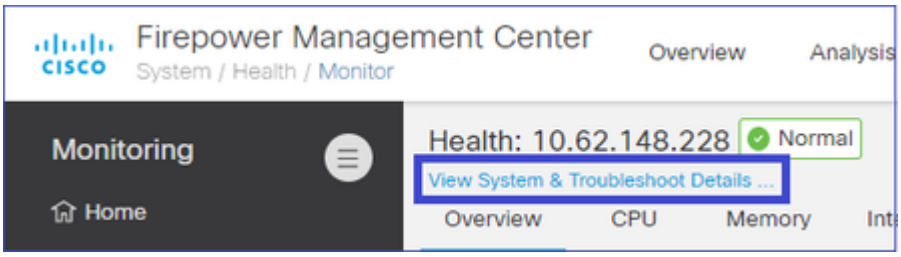

admin@firepower:~$

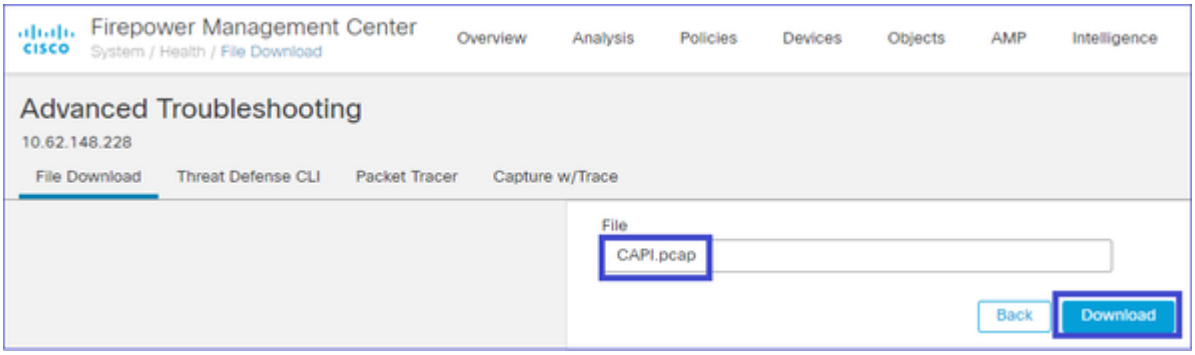
cd /mnt/disk0


admin@firepower:/mnt/disk0$
```

```
sudo cp CAPI.pcap /ngfw/var/common
```

Enfin, sur FMC, accédez à System > Health > Monitor section. Choisissez View System & Troubleshoot Details > Advanced Troubleshooting et récupérez le fichier de capture :





Supprimer une capture

Pour supprimer une capture de toutes les unités de cluster, utilisez cette commande :

<#root>

```
firepower#
cluster exec no capture CAPI

unit-1-1(LOCAL):**********************************************************
unit-2-1:**********************************************************
unit-3-1:**********************************************************
```

Flux déchargés

Sur les modèles FP41xx/FP9300, les flux peuvent être transférés vers HW Accelerator de manière statique (par exemple, les règles Fastpath) ou dynamique. Pour plus d'informations sur le déchargement de flux, consultez ce document : Clarifier les actions de règle de contrôle d'accès FTD.

Si un flux est déchargé, seuls quelques paquets passent par le plan de données FTD. Le reste est géré par l'accélérateur matériel (carte réseau intelligente).

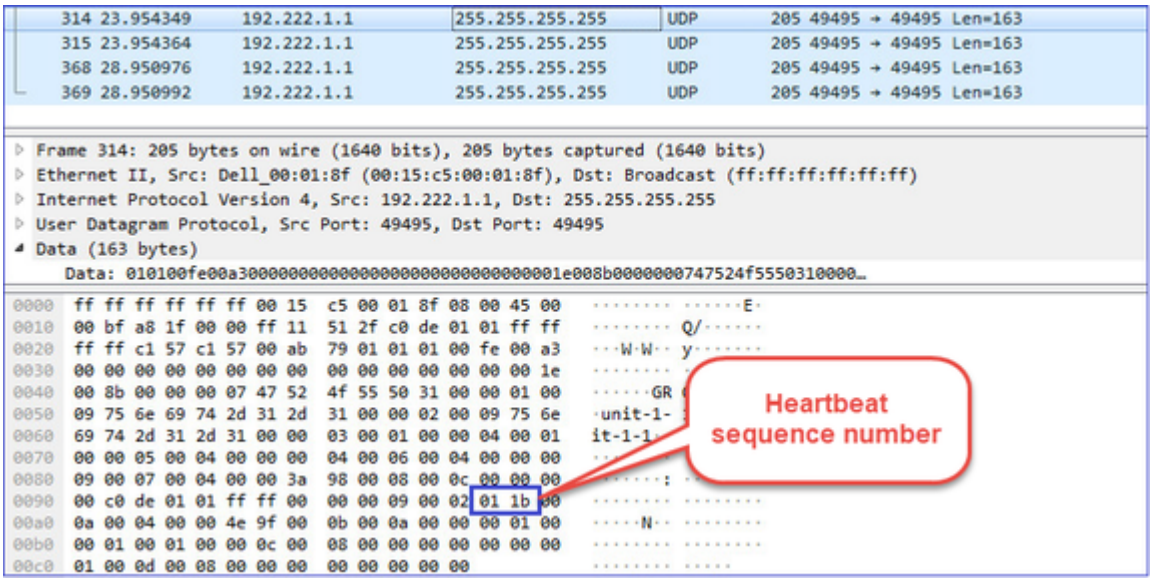
Du point de vue de la capture, cela signifie que si vous activez uniquement les captures au niveau du plan de données FTD, vous ne voyez pas tous les paquets qui passent par le périphérique. Dans ce cas, vous devez également activer les captures au niveau du châssis FXOS.

Messages CCL (Cluster Control Link)

Si vous effectuez une capture sur la CCL, vous remarquerez que les unités de cluster échangent différents types de messages. Ceux qui vous intéressent sont :

Protocol	Description
UDP 49495	Pulsations de cluster (keepalives) · Diffusion L3 (255.255.255.255) · Ces paquets sont envoyés par chaque unité de cluster à 1/3 de la valeur de temps d'attente du contrôle d'intégrité. · Notez que tous les paquets UDP 49495 vus dans la capture ne sont pas des pulsations · Les battements de coeur contiennent un numéro d'ordre.
UDP 4193	Messages de chemin de données du protocole de contrôle de cluster · Monodiffusion · Ces paquets contiennent des informations (métadonnées) sur le propriétaire du flux, le directeur, le propriétaire de la sauvegarde, etc. Exemples : · Un message « cluster add » est envoyé par le propriétaire au directeur lorsqu'un nouveau flux est créé · Un message « cluster delete » est envoyé par le propriétaire au directeur lorsqu'un flux est interrompu
paquets de données	Paquets de données qui appartiennent aux différents flux de trafic qui traversent le cluster

Pulsation du cluster :



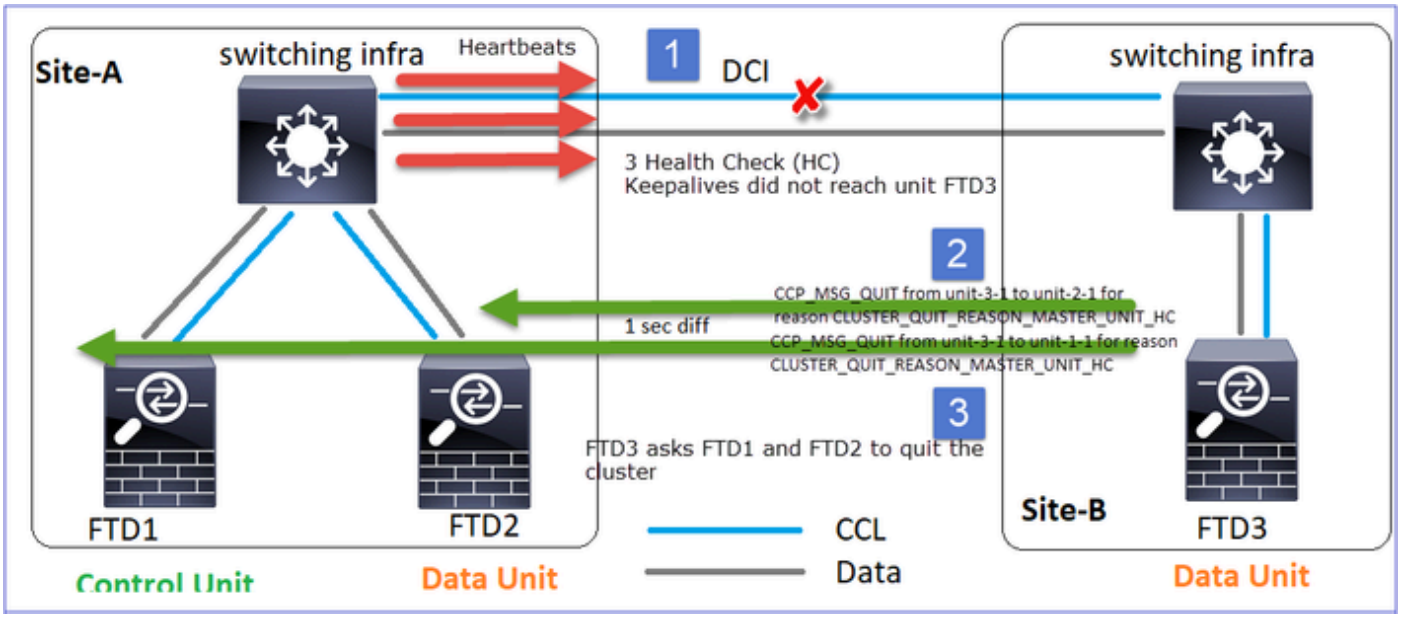
Messages CCP (Cluster Control Point)

Outre les messages de pulsation, un certain nombre de messages de contrôle de cluster sont échangés via la CCL dans des scénarios spécifiques. Certains d'entre eux sont des messages de monodiffusion tandis que d'autres sont des diffusions.

CLUSTER_QUIT_REASON_PRIMARY_UNIT_HC

Chaque fois qu'une unité perd 3 messages de pulsation consécutifs en provenance du noeud de contrôle, elle génère un message CLUSTER_QUIT_REASON_PRIMARY_UNIT_HC sur la CCL. Ce message:

- Est une monodiffusion.
- Il est envoyé à chacune des unités avec un intervalle d'une seconde.
- Lorsqu'une unité reçoit ce message, quitte le cluster (DISABLED) et se reconnecte.

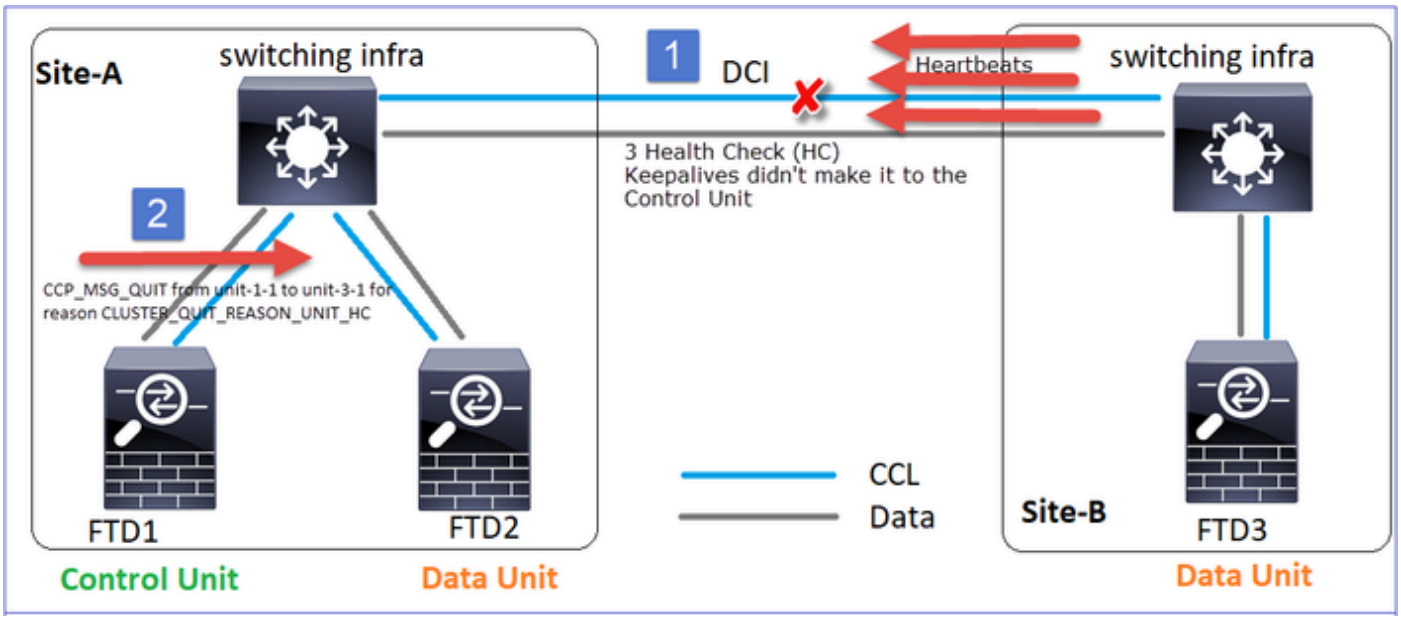


Q. Quelle est la fonction de CLUSTER_QUIT_REASON_PRIMARY_UNIT_HC ?

A. Du point de vue de l'unité-3-1 (Site-B), il perd la connexion à l'unité-1-1 et à l'unité-2-1 du site A, donc il doit les retirer de sa liste de membres dès que possible, sinon, il peut avoir un paquet perdu si l'unité-2-1 est toujours dans sa liste de membres et que l'unité-2-1 se trouve être un directeur d'une connexion, et la requête de flux à l'unité-2-1 échoue.

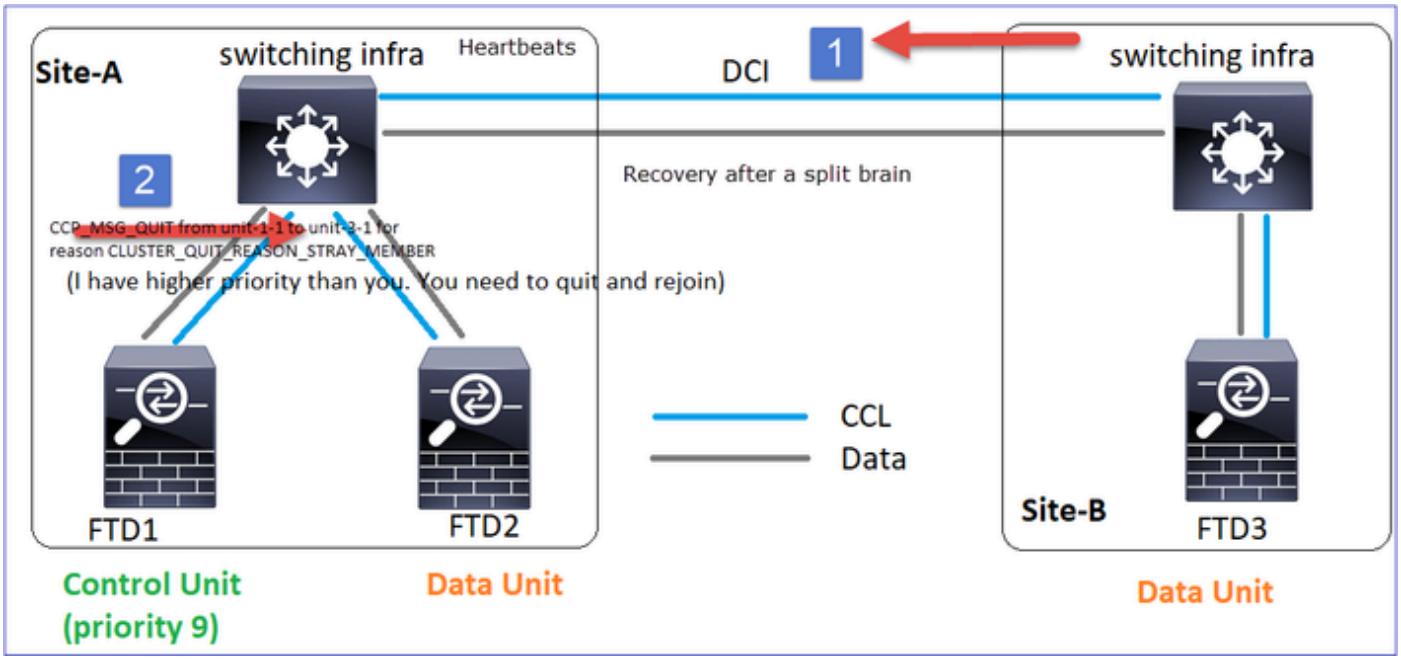
CLUSTER_QUIT_REASON_UNIT_HC

Chaque fois que le noeud de contrôle perd 3 messages de pulsation consécutifs à partir d'un noeud de données, il envoie un message CLUSTER_QUIT_REASON_UNIT_HC sur la CCL. Ce message est en monodiffusion.



MEMBRE_PARCOURS_RAISON_ARRÊT_GRAPPE

Lorsqu'une partition partagée se reconnecte à une partition homologue, le nouveau noeud de données est traité comme un membre parasite par l'unité de contrôle dominante et reçoit un message CCP quit avec la raison CLUSTER_QUIT_REASON_STRAY_MEMBER.



ABANDON_MEMBRE_QUIT_CLUSTER

Message de diffusion généré par un noeud de données et envoyé sous forme de diffusion. Une fois qu'une unité reçoit ce message, passe à l'état DISABLED. En outre, la réjointure automatique ne démarre pas :

```
<#root>

firepower#

show cluster info trace | include DROPOUT

Nov 04 00:22:54.699 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-1-1 for reason

CLUSTER_QUIT_MEMBER_DROPOUT

Nov 04 00:22:53.699 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-2-1 for reason

CLUSTER_QUIT_MEMBER_DROPOUT
```

L'historique du cluster indique :

```
<#root>

PRIMARY       DISABLED      Received control message DISABLE (

member dropout announcement

)
```

Mécanisme de vérification de l'état de santé des clusters (HC)

Points principaux:

- Chaque unité de cluster envoie une pulsation toutes les 1/3 de la valeur de temps d'attente du contrôle d'intégrité à toutes les autres unités (diffusion 255.255.255.255) et utilise le port UDP 49495 comme transport sur la CCL.
- Chaque unité de grappe suit indépendamment une unité sur deux avec un compteur d'interrogation et une valeur de comptage d'interrogation.
- Si une unité de cluster ne reçoit aucun paquet (pulsation ou paquet de données) d'une unité homologue de cluster au cours d'un intervalle de pulsation, elle augmente la valeur du nombre d'interrogations.
- Lorsque la valeur du nombre d'interrogations pour une unité homologue de cluster devient 3, l'homologue est considéré comme inactif.
- Chaque fois qu'un battement de coeur est reçu, son numéro de séquence est vérifié et dans le cas où la différence avec le battement de coeur précédemment reçu est différente de 1, le compteur d'abandon de battement de coeur augmente en conséquence.

- Si le compteur du nombre d'interrogations d'un homologue de cluster est différent de 0 et qu'un paquet est reçu par l'homologue, le compteur est réinitialisé à une valeur 0.

Utilisez cette commande pour vérifier les compteurs d'intégrité du cluster :

<#root>

```
firepower#
```

show cluster info health details

```
------------------------------------------------------------------------------
|                Unit (ID)| Heartbeat| Heartbeat|   Average|   Maximum|      Poll|
|                         |     count|     drops|  gap (ms)| slip (ms)|     count|
------------------------------------------------------------------------------
|            unit-2-1 ( 1)|       650|         0|      4999|         1|         0|
|            unit-3-1 ( 2)|       650|         0|      4999|         1|         0|
------------------------------------------------------------------------------
```

Description des principales colonnes :

Colonne	Description
Unité (ID)	ID de l'homologue de cluster distant.
Nombre de pulsations	Nombre de pulsations reçues de l'homologue distant sur la CCL.
Les pulsations chutent	Nombre de pulsations manquées. Ce compteur est calculé en fonction du numéro de séquence de pulsation reçu.
Écart moyen	Intervalle de temps moyen des pulsations reçues.
Nombre de sondages	Lorsque ce compteur passe à 3, l'unité est retirée de la grappe. L'intervalle de requête d'interrogation est identique à l'intervalle de pulsation, mais s'exécute indépendamment.

Pour réinitialiser les compteurs, utilisez cette commande :

<#root>

```
firepower#
```

clear cluster info health details

Q. Comment vérifier la fréquence des battements de coeur ?

A. Vérifiez la valeur moyenne de l'écart :

<#root>

```
firepower#
show cluster info health details

-------------------------------------------------------------------------------
|                Unit (ID)| Heartbeat| Heartbeat|
Average
|   Maximum|      Poll|
|                         |     count|     drops|
gap (ms)
| slip (ms)|     count|
-------------------------------------------------------------------------------
|            unit-2-1 ( 1)|      3036|         0|
999
|         1|         0|
-------------------------------------------------------------------------------
```

Q. Comment pouvez-vous modifier le temps d'attente du cluster sur le FTD ?

A. Utilisez FlexConfig.

Q. Qui devient le noeud de contrôle après une scission du cerveau ?

A. L'unité ayant la priorité la plus élevée (numéro le plus bas) :

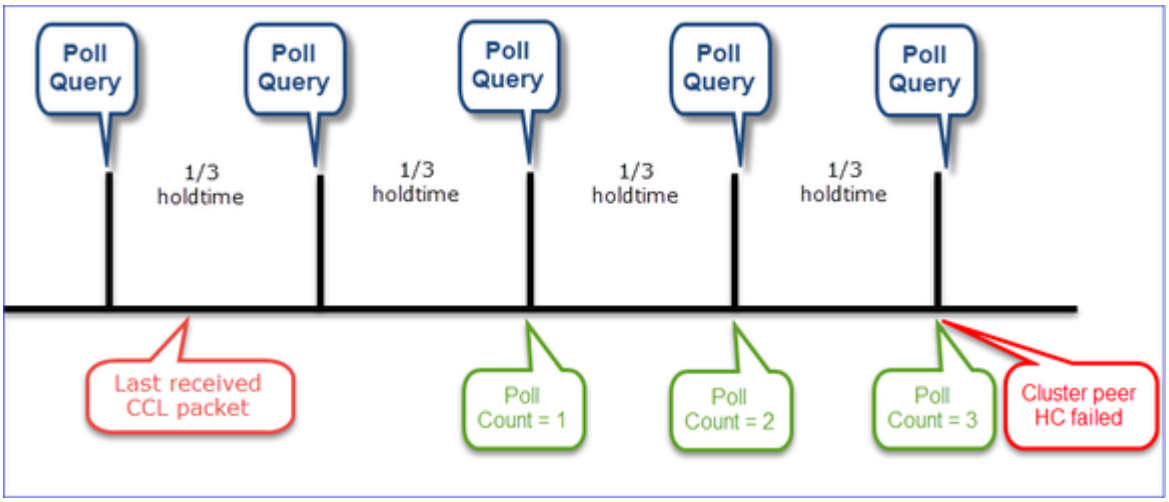
<#root>

```
firepower#
show run cluster | include priority

priority 9
```

Consultez le scénario de défaillance HC 1 pour plus de détails.

Visualisation du mécanisme HC du cluster



Minuteurs indicatifs : Les valeurs min et max dépendent de la dernière arrivée du paquet CCL reçu.

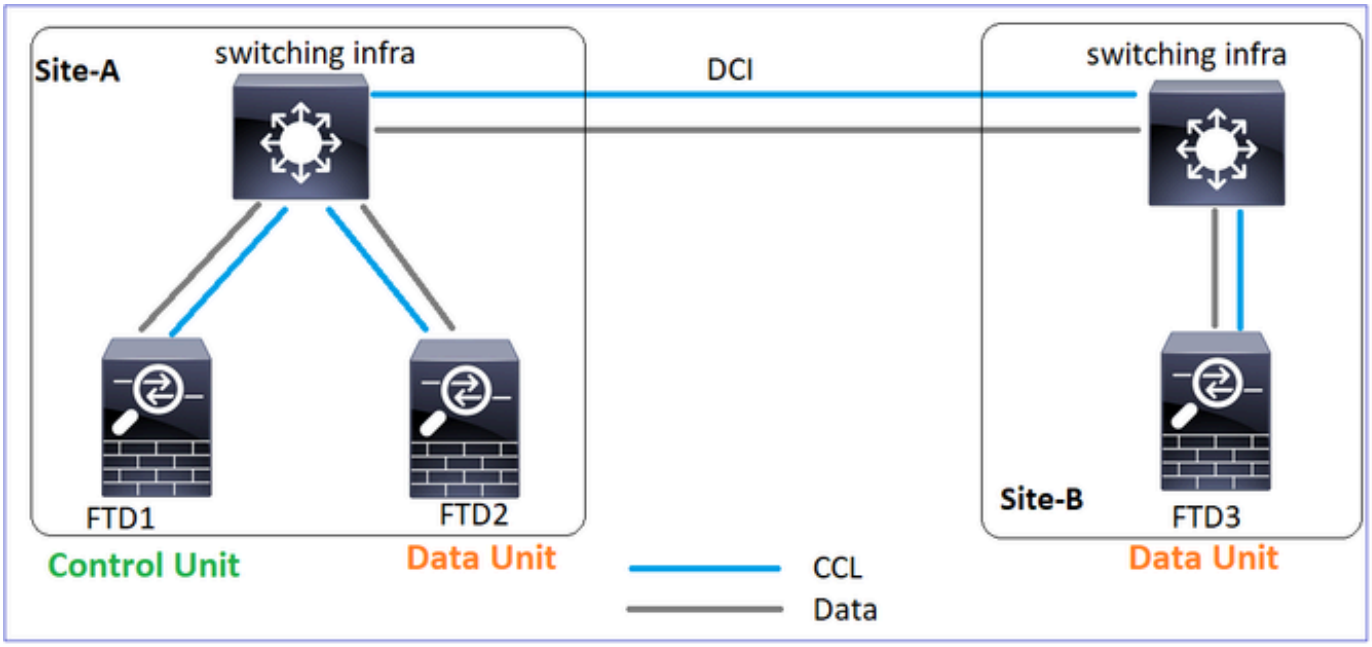
Temps d'attente	Vérification de requête de sondage (fréquence)	Temps de détection minimal	Temps de détection max.
3 s (par défaut)	~1 s	~3,01 s	~3,99 s
4 sec	~1,33 s	~4,01 s	~5,32 s
5 sec	~1,66 s	~5,01 s	~6,65 s
6 sec	~2 s	~6,01 s	~7,99 s
7 sec	~2,33 s	~7,01 s	~9,32 s
8 sec	~2,66 s	~8,01 s	~10,65 s

Scénarios de défaillance HC de cluster

Les objectifs de cette section sont de démontrer :

- Différents scénarios de défaillance HC de cluster.
- Comment corréler les différents journaux et les résultats des commandes.

Topologie



Configuration du cluster :

Unité-1-1	Unité-2-1
cluster group GROUP1 key ***** local-unit unit-1-1 cluster-interface Port-channel48 ip 10.17.1.1 255.255.0.0 priority 9 health-check holdtime 3 health-check data-interface auto-rejoin 3 5 2 health-check cluster-interface auto-rejoin unlimited 5 1 health-check system auto-rejoin 3 5 2 health-check monitor-interface debounce-time 500 site-id 1 enable	cluster group GROUP key ***** local-unit unit-2- cluster-interface priority 17 health-check holdt health-check data- health-check clust health-check syste health-check monit site-id 1 enable

État du cluster :

Unité-1-1	Unité-2-1
<#root> firepower# **show cluster info**	<#root> firepower# **show cluster info**

```
Cluster GROUP1: On
    Interface mode: spanned

This is "unit-1-1" in state PRIMARY

        ID        : 0
        Site ID   : 1
        Version   : 9.12(2)33
        Serial No.: FCH22247LNK
        CCL IP    : 10.17.1.1
        CCL MAC   : 0015.c500.018f
        Last join : 20:25:36 UTC Nov 1 2020
        Last leave: 20:25:28 UTC Nov 1 2020
Other members in the cluster:

Unit "unit-3-1" in state secondary

        ID        : 1
        Site ID   : 2
        Version   : 9.12(2)33
        Serial No.: FCH22247MKJ
        CCL IP    : 10.17.3.1
        CCL MAC   : 0015.c500.038f
        Last join : 20:58:45 UTC Nov 1 2020
        Last leave: 20:58:37 UTC Nov 1 2020

Unit "unit-2-1" in state SECONDARY

        ID        : 2
        Site ID   : 1
        Version   : 9.12(2)33
        Serial No.: FCH23157Y9N
        CCL IP    : 10.17.2.1
        CCL MAC   : 0015.c500.028f
        Last join : 20:44:45 UTC Nov 1 2020
        Last leave: 20:44:38 UTC Nov 1 2020
```

```
Cluster GROUP1: On
    Interface mode: spanned

This is "unit-2-1" in state SECONDARY

        ID        : 2
        Site ID   : 1
        Version   : 9.12(2)33
        Serial No.: FCH23157Y9N
        CCL IP    : 10.17.2.1
        CCL MAC   : 0015.c500.028f
        Last join : 20:44:46 UTC N
        Last leave: 20:44:38 UTC N
Other members in the cluster:

 Unit "unit-1-1" in state PRIMARY

        ID        : 0
        Site ID   : 1
        Version   : 9.12(2)33
        Serial No.: FCH22247LNK
        CCL IP    : 10.17.1.1
        CCL MAC   : 0015.c500.018f
        Last join : 20:25:36 UTC N
        Last leave: 20:25:28 UTC N

Unit "unit-3-1" in state SECONDARY

        ID        : 1
        Site ID   : 2
        Version   : 9.12(2)33
        Serial No.: FCH22247MKJ
        CCL IP    : 10.17.3.1
        CCL MAC   : 0015.c500.038f
        Last join : 20:58:45 UTC N
        Last leave: 20:58:37 UTC N
```

Scénario 1 - Perte de communication CCL pour ~4+ secondes dans les deux directions

Avant la panne :

FTD1	FTD2	FTD3
Site-A	Site-A	Site-B

Noeud de contrôle	Noeud Données	Noeud Données

Après la restauration (aucune modification des rôles d'unité) :

FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

Analyse

Échec (communication CCL perdue).



Le message de console du plan de données sur l'unité 3-1 :

<#root>

```
firepower#
WARNING: dynamic routing is not supported on management interface when cluster interface-mode is 'spann
If dynamic routing is configured on any management interface, please remove it.


Cluster unit unit-3-1 transitioned from SECONDARY to PRIMARY


Cluster disable is performing cleanup..done.
All data interfaces have been shutdown due to clustering being disabled.
To recover either enable clustering or remove cluster group configuration.
```

Journaux de suivi de cluster Unit-1-1 :

<#root>

```
firepower#

show cluster info trace | include unit-3-1
```

```
Nov 02 09:38:14.239 [INFO]Notify chassis de-bundle port for blade unit-3-1, stack 0x000055a8918307fb 0x
Nov 02 09:38:14.239 [INFO]FTD - CD proxy received state notification (DISABLED) from unit unit-3-1
Nov 02 09:38:14.239
```

[DBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CLUSTER_QUIT_MEMBER_DRO

```
Nov 02 09:38:14.239 [INFO]Notify chassis de-bundle port for blade unit-3-1, stack 0x000055a8917eb596 0x
Nov 02 09:38:14.239
```

[DBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CLUSTER_QUIT_REASON_UN

```
Nov 02 09:38:14.239 [CRIT]Received heartbeat event 'SECONDARY heartbeat failure' for member unit-3-1 (I
```

Split-brain :

Unité-1-1	Unité-2-1
<#root> firepower# **show cluster info** Cluster GROUP1: On Interface mode: spanned **This is "unit-1-1" in state PRIMARY** ID : 0 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 20:25:36 UTC Nov 1 2020 Last leave: 20:25:28 UTC Nov 1 2020 Other members in the cluster: Unit "unit-2-1" in state SECONDARY ID : 2 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028f Last join : 20:44:45 UTC Nov 1 2020 Last leave: 20:44:38 UTC Nov 1 2020	<#root> firepower# **show cluster info** Cluster GROUP1: On Interface mode: spanned This is "unit-2-1" in state SE ID : 2 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028f Last join : 20:44:46 UTC N Last leave: 20:44:38 UTC N Other members in the cluster: **Unit "unit-1-1" in state PRIMARY** ID : 0 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 20:25:36 UTC N Last leave: 20:25:28 UTC N

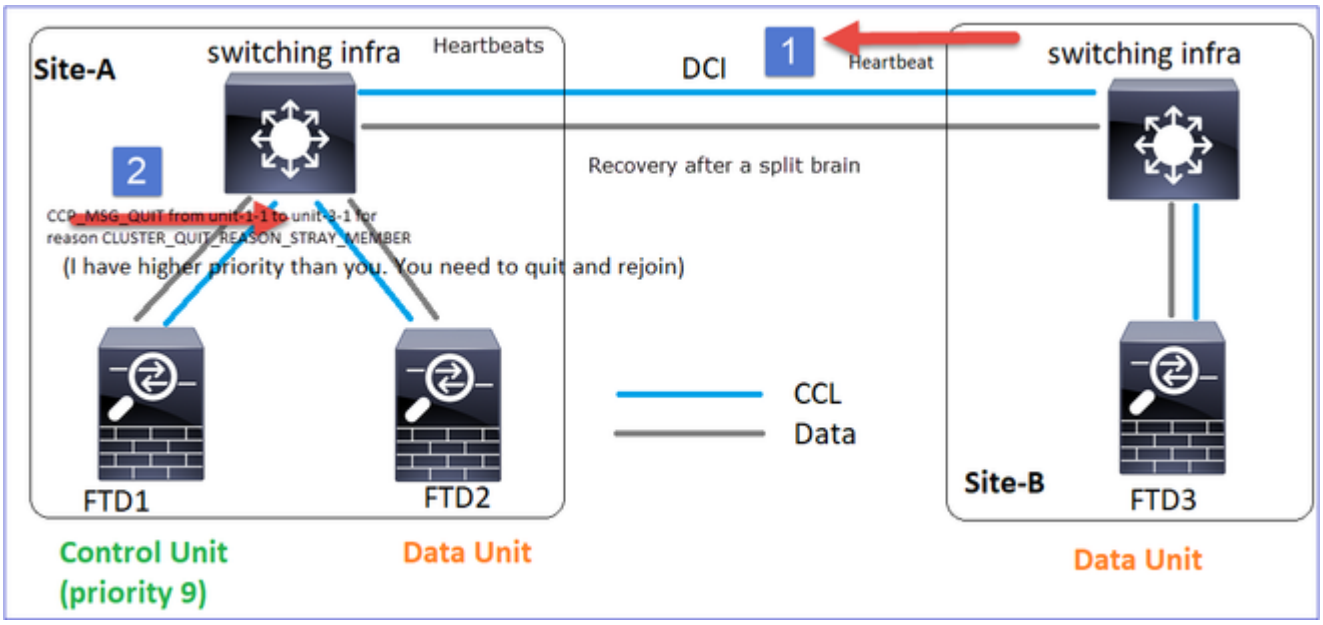
Historique du cluster :

Unité-1-1	Unité-2-1	Unité-3-1
Aucun événement	Aucun événement	<#root> 09:38:16 UTC Nov 2 2020 **SECONDARY PRIMARY_POST_CONFIG Primary rel** 09:38:17 UTC Nov 2 2020 PRIMARY_POST_CONFIG Primary Primary post co

Restauration des communications CCL

Unit-1-1 détecte le noeud de contrôle actuel et, comme unit-1-1 a une priorité plus élevée, envoie à unit-3-1 un message CLUSTER_QUIT_REASON_STRAY_MEMBER pour déclencher un nouveau processus de sélection. En fin de compte, l'unité 3-1 se joint à nouveau en tant que noeud de données.

Lorsqu'une partition partagée se reconnecte à une partition homologue, le noeud de données est traité comme un membre parasite par le noeud de contrôle dominant et reçoit un message CCP quit avec une raison de CLUSTER_QUIT_REASON_STRAY_MEMBER.



<#root>

Unit-3-1 console logs show:

Cluster unit unit-3-1 transitioned from PRIMARY to DISABLED

The 3DES/AES algorithms require a Encryption-3DES-AES activation key.

```
Detected Cluster Primart.


Beginning configuration replication from Primary.
WARNING: Local user database is empty and there are still 'aaa' commands for 'LOCAL'.
..
Cryptochecksum (changed): a9ed686f 8e2e689c 2553a104 7a2bd33a
End configuration replication from Primary.


Cluster unit unit-3-1 transitioned from DISABLED to SECONDARY
```

Les deux unités (unit-1-1 et unit-3-1) affichent dans leurs journaux de cluster :

```
<#root>

firepower#

show cluster info trace | include retain


Nov 03 21:20:23.019 [CRIT]Found a split cluster with both unit-1-1 and unit-3-1 as primary units. Prima
Nov 03 21:20:23.019 [CRIT]Found a split cluster with both unit-1-1 and unit-3-1 as primary units. Prima
```

Il y a aussi des messages syslog générés pour le split-brain :

```
<#root>

firepower#

show log | include 747016


Nov 03 2020 21:20:23: %FTD-4-747016: Clustering: Found a split cluster with both unit-1-1 and unit-3-1 
Nov 03 2020 21:20:23: %FTD-4-747016: Clustering: Found a split cluster with both unit-1-1 and unit-3-1 
```

Historique du cluster :

Unité-1-1	Unité-2-1	Unité-3-1
Aucun événement	Aucun événement	<#root> 09:47:33 UTC Nov 2 2020 **Primary DISABLED Detected a splitted cluster** 09:47:38 UTC Nov 2 2020

		DISABLED ELECTION Enabled from CLI 09:47:38 UTC Nov 2 2020 ELECTION SECONDARY_COLD Received cluster c 09:47:38 UTC Nov 2 2020 SECONDARY_COLD SECONDARY_APP_SYNC Client progres 09:48:18 UTC Nov 2 2020 SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY appl 09:48:29 UTC Nov 2 2020 SECONDARY_CONFIG SECONDARY_FILESYS Configuration 09:48:30 UTC Nov 2 2020 SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progres 09:48:54 UTC Nov 2 2020 SECONDARY_BULK_SYNC **SECONDARY** Client progression done

Scénario 2 - Perte de communication CCL d'environ 3 à 4 secondes dans les deux sens

Avant la panne :

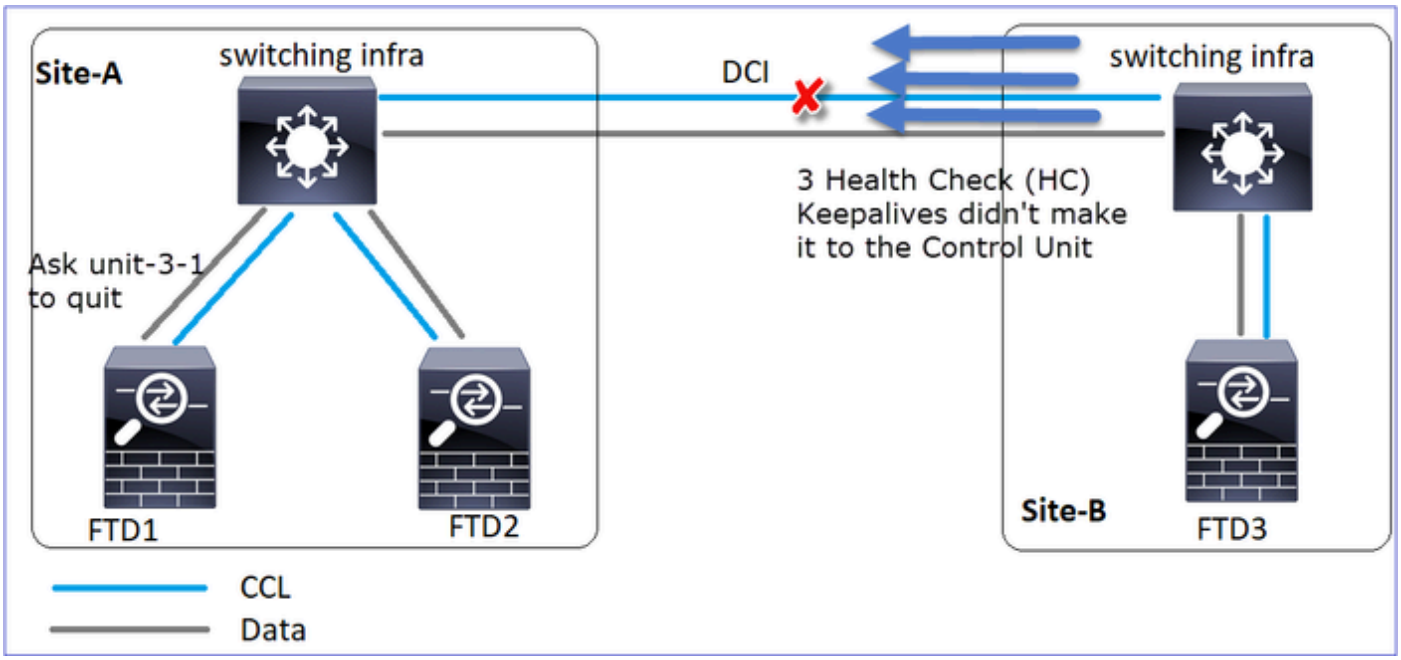
FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

Après la restauration (aucune modification des rôles d'unité) :

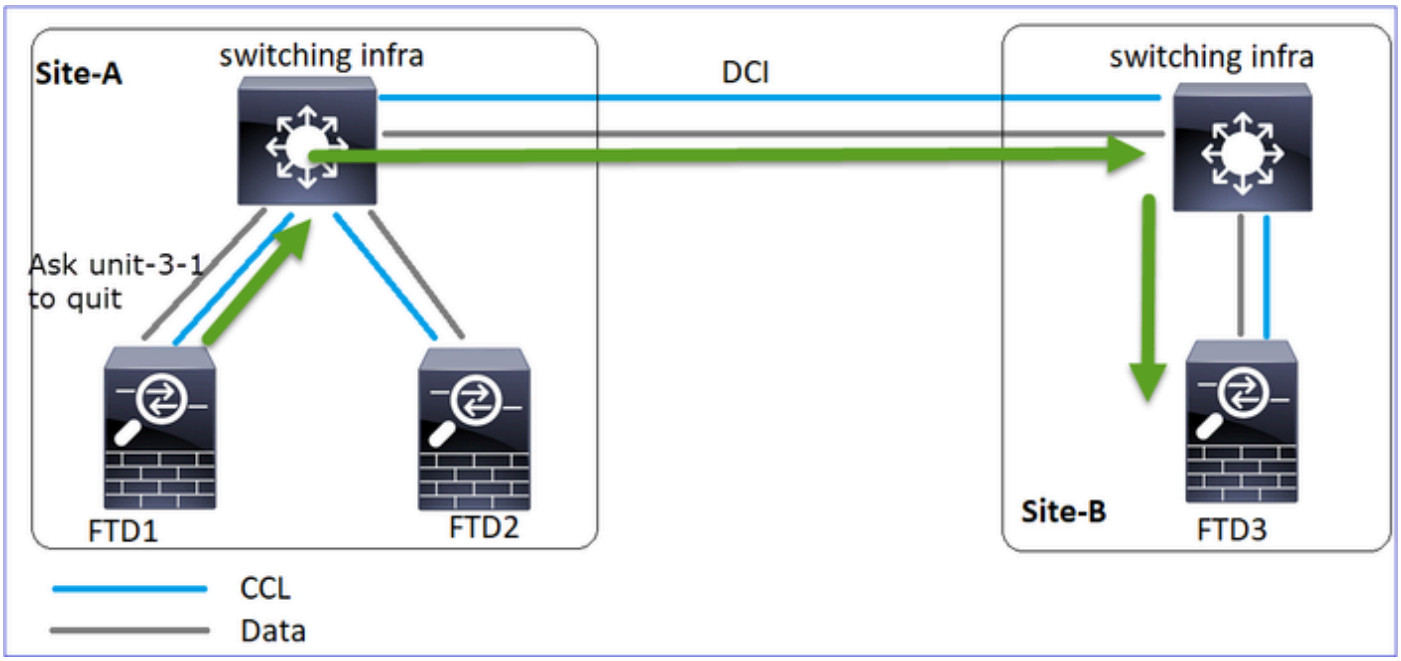
FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

Analyse

Événement 1 : Le noeud de contrôle perd 3 HC de l'unité-3-1 et envoie un message à l'unité-3-1 pour quitter la grappe.



Événement 2 : La CCL s'est rétablie très rapidement et le message CLUSTER_QUIT_REASON_STRAY_MEMBER du noeud de contrôle est arrivé sur le côté distant. Unit-3-1 passe directement en mode DISABLED et il n'y a pas de split-brain



Sur l'unité 1-1 (contrôle), vous voyez :

<#root>

```
firepower#
Asking SECONDARY unit unit-3-1 to quit because it failed unit health-check.
```

Forcing stray member unit-3-1 to leave the cluster

Sur l'unité 3-1 (noeud de données), vous voyez :

```
<#root>

firepower#

Cluster disable

 is performing cleanup..done.
All data interfaces have been shutdown due to clustering being disabled. To recover either enable clust

Cluster unit unit-3-1 transitioned from SECONDARY to DISABLED
```

L'unité de cluster 3-1 est passée à l'état DISABLED et, une fois la communication CCL restaurée, elle rejoint à nouveau le réseau en tant que noeud de données :

```
<#root>

firepower#

show cluster history

20:58:40 UTC Nov 1 2020

SECONDARY                 DISABLED              Received control message DISABLE (stray member)

20:58:45 UTC Nov 1 2020
DISABLED              ELECTION              Enabled from CLI
20:58:45 UTC Nov 1 2020
ELECTION              SECONDARY_COLD            Received cluster control message
20:58:45 UTC Nov 1 2020
SECONDARY_COLD            SECONDARY_APP_SYNC        Client progression done
20:59:33 UTC Nov 1 2020
SECONDARY_APP_SYNC        SECONDARY_CONFIG          SECONDARY application configuration sync done
20:59:44 UTC Nov 1 2020
SECONDARY_CONFIG          SECONDARY_FILESYS         Configuration replication finished
20:59:45 UTC Nov 1 2020
SECONDARY_FILESYS         SECONDARY_BULK_SYNC       Client progression done
21:00:09 UTC Nov 1 2020

SECONDARY_BULK_SYNC       SECONDARY

                Client progression done
```

Scénario 3 - Perte de communication CCL d'environ 3 à 4 secondes dans les deux sens

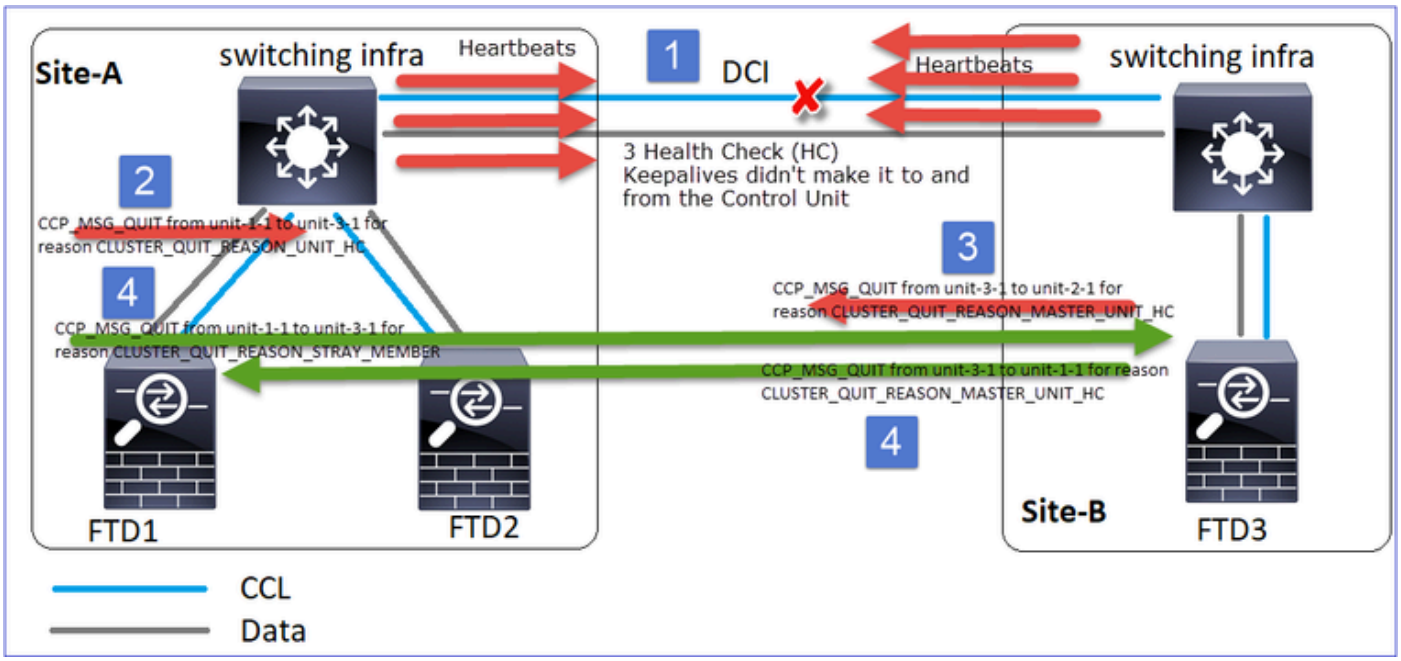
Avant la panne :

FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

Après la restauration (le noeud de contrôle a été modifié) :

FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud Données	Noeud de contrôle	Noeud Données

Analyse



1. CCL tombe en panne.
2. Unit-1-1 n'obtient pas 3 messages HC de unit-3-1 et envoie un message QUIT à unit-3-1. Ce message n'atteint jamais unit-3-1.
3. Unit-3-1 envoie un message QUIT à unit-2-1. Ce message n'atteint jamais unit-2-1.

CCL récupère.

4. L'unité 1-1 voit que l'unité 3-1 s'est annoncée comme noeud de contrôle et envoie le message QUIT_REASON_STRAY_MEMBER à l'unité 3-1. Une fois que l'unité 3-1 reçoit ce message, elle passe à l'état DISABLED. En même temps, l'unité-3-1 envoie un message

QUIT_REASON_PRIMARY_UNIT_HC à l'unité-1-1 et lui demande de quitter. Une fois que l'unité 1-1 reçoit ce message, il passe à l'état DISABLED.

Historique du cluster :

Unité-1-1
<#root> 19:53:09 UTC Nov 2 2020 **PRIMARY DISABLED** Received control message DISABLE (primary unit health check failure) 19:53:13 UTC Nov 2 2020 DISABLED ELECTION Enabled from CLI 19:53:13 UTC Nov 2 2020 ELECTION SECONDARY_COLD Received cluster control message 19:53:13 UTC Nov 2 2020 SECONDARY_COLD SECONDARY_APP_SYNC Client progression done 19:54:01 UTC Nov 2 2020 SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application configur 19:54:12 UTC Nov 2 2020 SECONDARY_CONFIG SECONDARY_FILESYS Configuration replication fini 19:54:13 UTC Nov 2 2020 SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done 19:54:37 UTC Nov 2 2020 SECONDARY_BULK_SYNC **SECONDARY** Client progression done

Scénario 4 - Perte de communication CCL d'environ 3 à 4 secondes

Avant la panne :

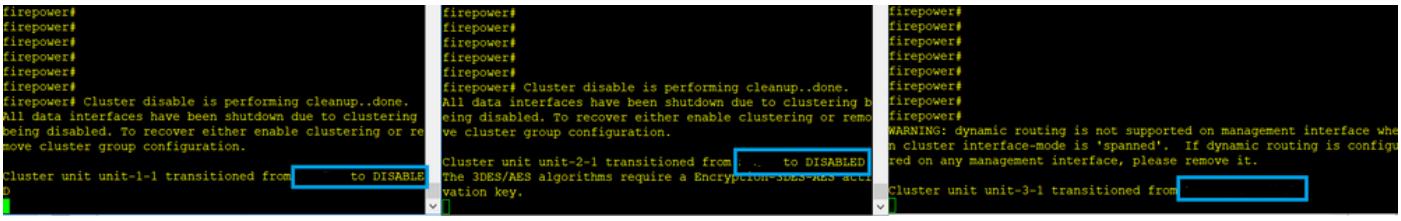
FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

Après la restauration (le noeud de contrôle a changé de site) :

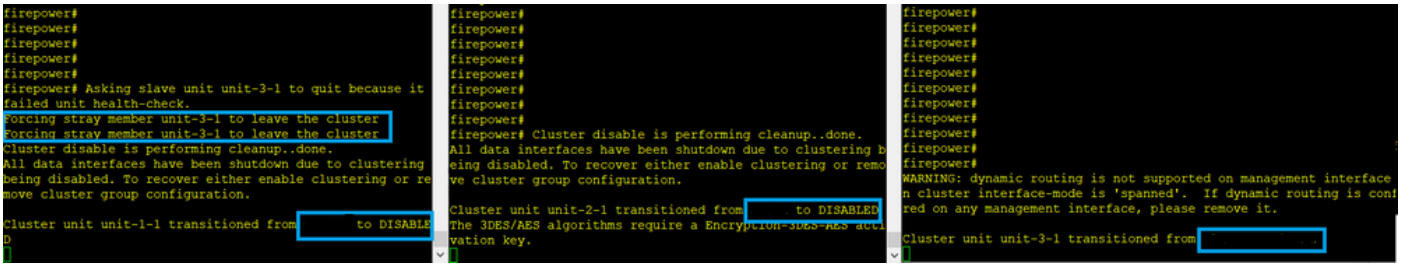
FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud Données	Noeud Données	Noeud de contrôle

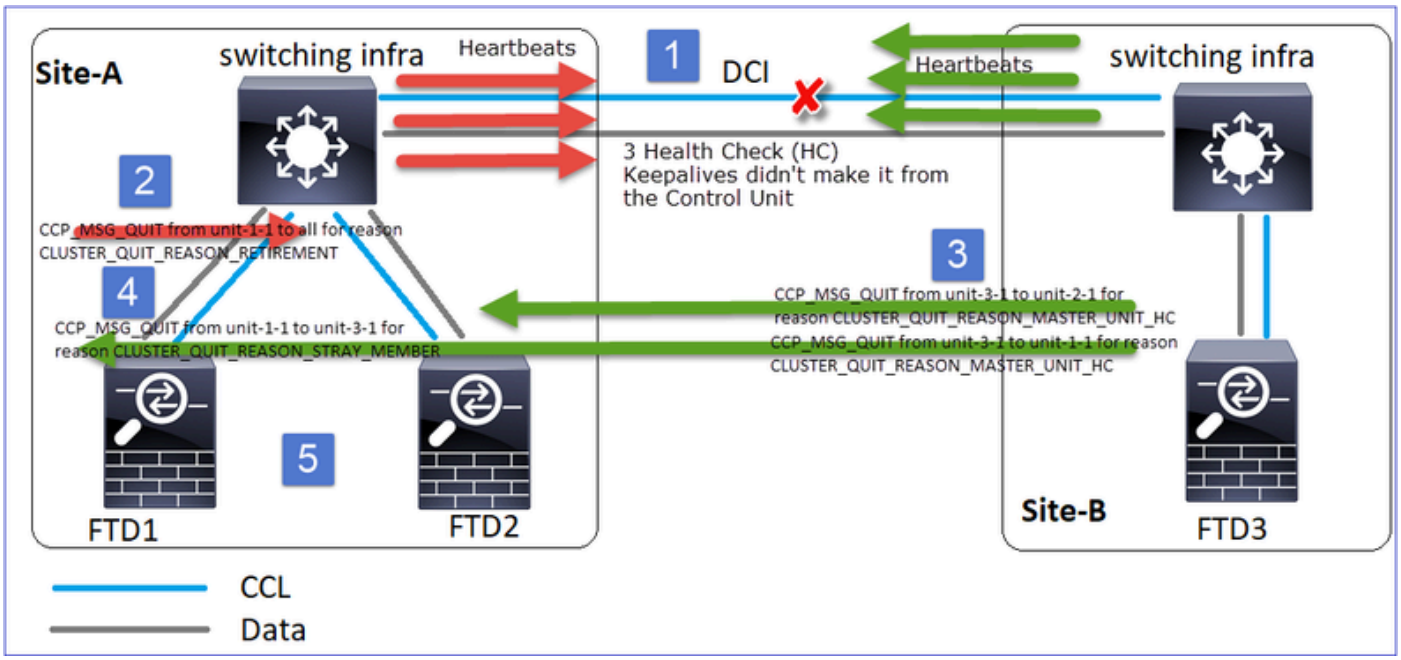
Analyse

L'échec



Un autre aspect de la même panne. Dans ce cas, l'unité-1-1 n'a pas non plus reçu 3 messages HC de l'unité-3-1, et une fois qu'elle a reçu un nouveau keepalive, elle a essayé de mettre l'unité-3-1 à la porte en utilisant un message STRAY, mais le message n'est jamais arrivé à l'unité-3-1 :





1. CCL devient unidirectionnelle pendant quelques secondes. L'unité 3-1 ne reçoit pas de messages 3 HC de l'unité 1-1 et devient un noeud de contrôle.
2. Unit-2-1 envoie un message CLUSTER_QUIT_REASON_RETIREMENT (diffusion).
3. Unit-3-1 envoie un message QUIT_REASON_PRIMARY_UNIT_HC à Unit-2-1. Unit-2-1 le reçoit et quitte le cluster.
4. Unit-3-1 envoie un message QUIT_REASON_PRIMARY_UNIT_HC à unit-1-1. Unit-1-1 le reçoit et quitte le cluster. CCL récupère.
5. Les unités 1-1 et 2-1 rejoignent le cluster en tant que noeuds de données.

Remarque : Si à l'étape 5 la CCL ne récupère pas, alors sur le site A le FTD1 devient le nouveau noeud de contrôle, et après la récupération de la CCL, il gagne la nouvelle sélection.

Messages Syslog sur unit-1-1 :

<#root>

```
firepower#
```

show log | include 747

```
Nov 03 2020 23:13:08: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:09: %FTD-4-747015: Clustering: Forcing stray member unit-3-1 to leave the cluster
Nov 03 2020 23:13:09: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:10: %FTD-4-747015: Clustering: Forcing stray member unit-3-1 to leave the cluster
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering:
```

State machine changed from state PRIMARY to DISABLED

```
Nov 03 2020 23:13:12: %FTD-7-747006: Clustering: State machine is at state DISABLED
Nov 03 2020 23:13:12: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MY_STATE (sta
```

```
Nov 03 2020 23:13:18: %FTD-6-747004: Clustering: State machine changed from state ELECTION to ONCALL
```

Journaux de suivi de cluster sur l'unité 1-1 :

<#root>

```
firepower#
```

show cluster info trace | include QUIT

```
Nov 03 23:13:10.789 [DBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 for reason CLUSTER_QUIT_R
Nov 03 23:13:10.769 [DBUG]
```

Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-1-1 for reason CLUSTER_QUIT_REASON_PRIMARY_UNIT_

```
Nov 03 23:13:10.769 [DBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason C
Nov 03 23:13:09.789 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-2-1 for reason CLUSTER_QUIT_REASO
Nov 03 23:13:09.769 [DBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason C
Nov 03 23:13:08.559 [DBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CL
Nov 03 23:13:08.559 [DBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason C
```

Messages Syslog sur unit-3-1 :

<#root>

```
firepower#
```

show log | include 747

```
Nov 03 2020 23:13:09: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:10: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering:
```

State machine changed from state SECONDARY to PRIMARY

```
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_FAST to PRIMA
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_DRAIN to PRIM
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_CONFIG to PRI
Nov 03 2020 23:13:10: %FTD-7-747006: Clustering: State machine is at state PRIMARY_POST_CONFIG
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_POST_CONFIG t
Nov 03 2020 23:13:10: %FTD-7-747006: Clustering:
```

State machine is at state PRIMARY

Historique du cluster

Unité-1-1	U

```
<#root>
23:13:13 UTC Nov 3 2020

PRIMARY       DISABLED        Received control message DISABLE
                                (primary unit health check failure)


23:13:18 UTC Nov 3 2020
DISABLED          ELECTION          Enabled from CLI
23:13:18 UTC Nov 3 2020
ELECTION          ONCALL            Received cluster control message
23:13:23 UTC Nov 3 2020
ONCALL            ELECTION          Received cluster control message
…
23:14:48 UTC Nov 3 2020
ONCALL            ELECTION          Received cluster control message
23:14:48 UTC Nov 3 2020
ELECTION          SECONDARY_COLD       Received cluster control message
23:14:48 UTC Nov 3 2020
SECONDARY_COLD       SECONDARY_APP_SYNC  Client progression done
23:15:36 UTC Nov 3 2020
SECONDARY_APP_SYNC  SECONDARY_CONFIG     SECONDARY application configuration
                                  sync done
23:15:48 UTC Nov 3 2020
SECONDARY_CONFIG     SECONDARY_FILESYS   Configuration replication finished
23:15:49 UTC Nov 3 2020
SECONDARY_FILESYS    SECONDARY_BULK_SYNC Client progression done
23:16:13 UTC Nov 3 2020
SECONDARY_BULK_SYNC

SECONDARY

            Client progression done
```

Scénario 5

Avant la panne :

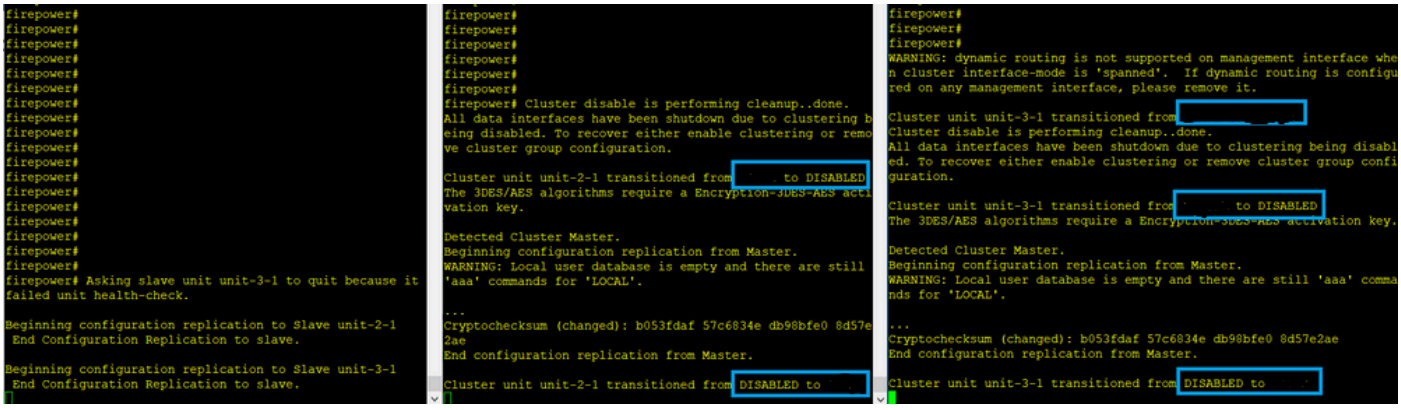
FTD1	FTD2	FTD3
Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

Après la restauration (aucune modification) :

FTD1	FTD2	FTD3

Site-A	Site-A	Site-B
Noeud de contrôle	Noeud Données	Noeud Données

L'échec :



Unit-3-1 a envoyé des messages QUIT à unit-1-1 et unit-2-1, mais en raison de problèmes de connectivité, seule l'unit-2-1 a reçu le message QUIT.

Journaux de suivi de cluster Unit-1-1 :

<#root>

firepower#

show cluster info trace | include QUIT

```
Nov 04 00:52:10.429 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-3-1 for reason CLUSTER_QUIT_REASO
Nov 04 00:51:47.059 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-2-1 for reason CLUSTER_QUIT_REASO
Nov 04 00:51:45.429 [DBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CL
Nov 04 00:51:45.429 [DBUG]Send CCP message to unit-3-1(1): CCP_MSG_QUIT from unit-1-1 to unit-3-1 for r
```

Journaux de suivi de cluster Unit-2-1 :

<#root>

firepower#

show cluster info trace | include QUIT

```
Nov 04 00:52:10.389 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-3-1 for reason CLUSTER_QUIT_REASO
Nov 04 00:51:47.019 [DBUG]Send CCP message to all: CCP_MSG_QUIT from unit-2-1 for reason CLUSTER_QUIT_R
Nov 04 00:51:46.999 [DBUG]
```

Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-2-1 for reason CLUSTER_QUIT_REASON_PRIMARY_UNIT_

```
Nov 04 00:51:45.389 [DBUG]Receive CCP message: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CLUSTE
```

Historique du cluster :

Unité-1-1	Unité-2-1
Aucun événement	<#root> 00:51:50 UTC Nov 4 2020 **SECONDARY DISABLED Received control message DISABLE** **(primary unit health check failure)** 00:51:54 UTC Nov 4 2020 DISABLED ELECTION Enabled from CLI 00:51:54 UTC Nov 4 2020 ELECTION SECONDARY_COLD Received cluster control messag 00:51:54 UTC Nov 4 2020 SECONDARY_COLD SECONDARY_APP_SYNC Client progression done 00:52:42 UTC Nov 4 2020 SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application confi sync done 00:52:54 UTC Nov 4 2020 SECONDARY_CONFIG SECONDARY_FILESYS Configuration replication f 00:52:55 UTC Nov 4 2020 SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done 00:53:19 UTC Nov 4 2020 SECONDARY_BULK_SYNC **SECONDARY** Client progression done

Établissement de la connexion du plan de données du cluster

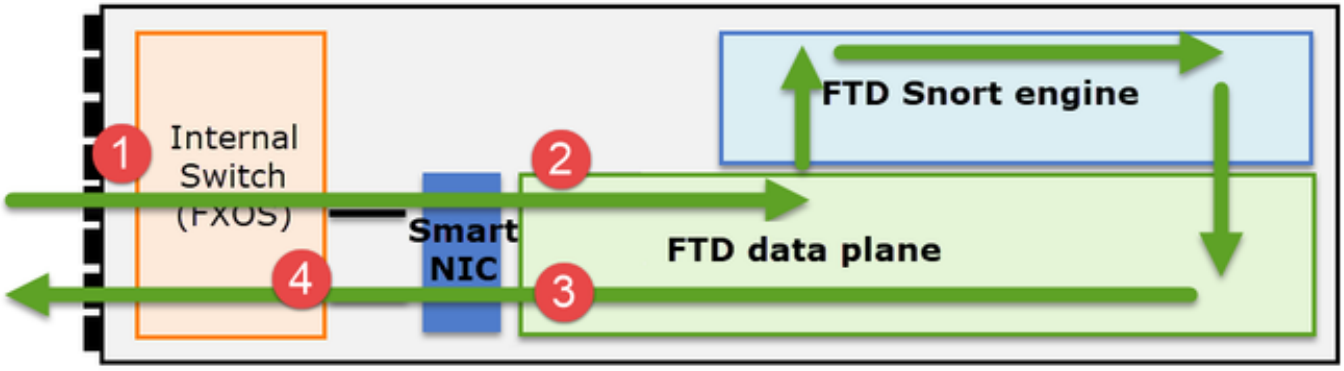
Points de capture NGFW

Le pare-feu de nouvelle génération offre des fonctionnalités de capture sur les points suivants :

- Commutateur interne du châssis (FXOS)
- moteur de plan de données FTD

- Moteur FTD Snort

Lorsque vous dépannez des problèmes de chemin de données sur un cluster, les points de capture utilisés dans la plupart des cas sont les captures de moteur de plan de données FXOS et FTD.



1. Capture d'entrée FXOS sur l'interface physique
2. Capture d'entrée FTD dans un moteur de plan de données
3. Capture de sortie FTD dans un moteur de plan de données
4. Capture d'entrée FXOS sur interface de fond de panier

Pour plus d'informations sur les captures NGFW, consultez ce document :

Notions de base sur les rôles de flux des unités

Les connexions peuvent être établies par le biais d'un cluster de plusieurs manières qui dépendent de facteurs tels que :

- Type de trafic (TCP, UDP, etc.)
- Algorithme d'équilibrage de charge configuré sur le commutateur adjacent
- Fonctionnalités configurées sur le pare-feu
- Conditions réseau (par exemple, fragmentation IP, retards réseau, etc.)

Rôle de flux	Description	Indicateur(s)
Propriétaire	Généralement, l'unité qui reçoit initialement la connexion	UIO
Direction	Unité qui gère les demandes de recherche de propriétaire provenant des redirecteurs.	O
Propriétaire de sauvegarde	Tant que le directeur n'est pas la même unité que le propriétaire, le directeur est	Y (si le directeur est également le propriétaire de la sauvegarde)

	également le propriétaire de secours. Si le propriétaire se choisit lui-même comme directeur, un propriétaire de sauvegarde distinct est choisi.	y (si le directeur n'est pas le propriétaire de la sauvegarde)
Transporteur	Unité qui transmet des paquets au propriétaire	z
Propriétaire du fragment	Unité qui gère le trafic fragmenté	-
Sauvegarde du châssis	Dans un cluster inter-châssis, lorsque les flux directeur/sauvegarde et propriétaire appartiennent aux unités du même châssis, une unité de l'un des autres châssis devient un directeur/sauvegarde secondaire. Ce rôle est spécifique aux clusters inter-châssis de la gamme Firepower 9300 avec plus d'1 lame.	w

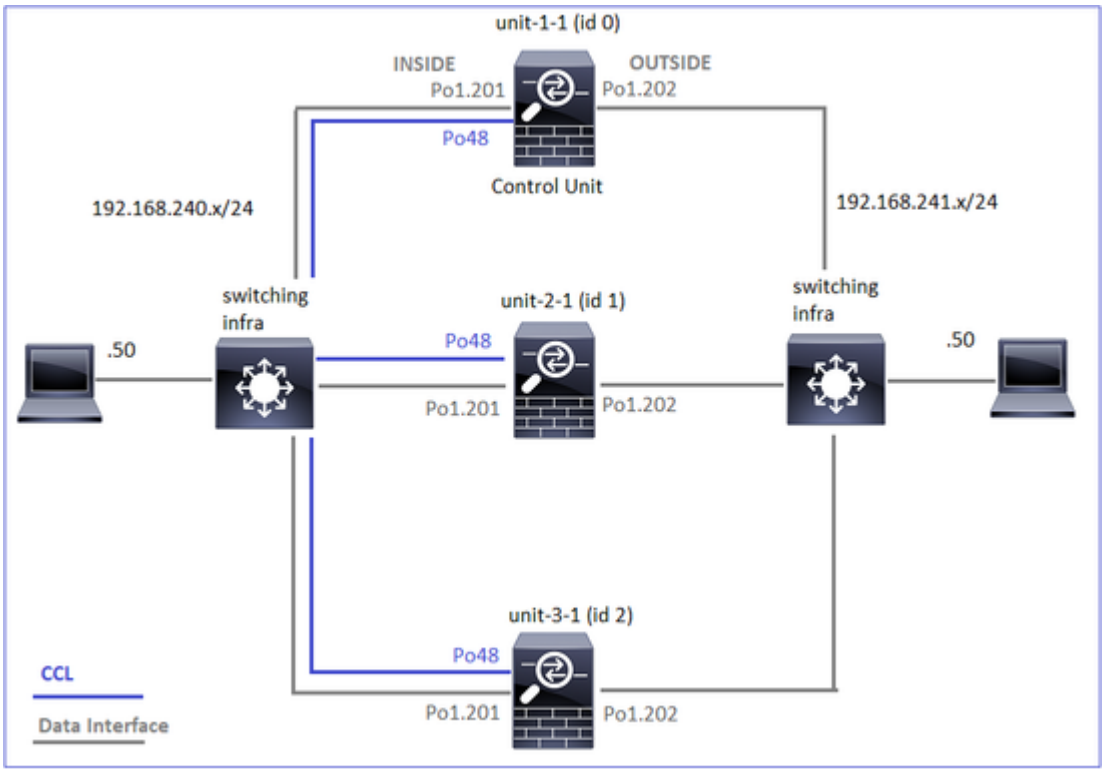
- Pour plus d'informations, consultez la section correspondante du Guide de configuration (voir les liens dans Informations connexes)
- Dans des scénarios spécifiques (voir la section des études de cas), certains indicateurs ne sont pas toujours affichés.

Études de cas sur Cluster Connection Establishment

La section suivante couvre diverses études de cas qui illustrent certaines des manières dont une connexion peut être établie par le biais d'un cluster. Les objectifs sont les suivants :

- Familiarisez-vous avec les différents rôles d'unité.
- Montrez comment les différentes sorties de commande peuvent être corrélées.

Topologie



Unités et ID de cluster :

Unité-1-1	Unité-2-1
<#root> Cluster GROUP1: On Interface mode: spanned **This is "unit-1-1" in state PRIMARY** **ID : 0** Site ID : 1 Version : 9.15(1) Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 02:24:43 UTC Nov 27 2020 Last leave: N/A	<#root> **Unit "unit-2-1" in state SECON** **ID : 1** Site ID : 1 Version : 9.15(1) Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028 Last join : 02:04:19 UTC Last leave: N/A

Captures de cluster activées :

```
cluster exec cap CAPI int INSIDE buffer 33554432 match tcp host 192.168.240.50 host 192.168.241.50 eq 8
```

```
cluster exec cap CAPO int OUTSIDE buffer 33554432 match tcp host 192.168.240.50 host 192.168.241.50 eq
cluster exec cap CAPI_RH reinject-hide int INSIDE buffer 33554432 match tcp host 192.168.240.50 host 19
cluster exec cap CAPO_RH reinject-hide int OUTSIDE buffer 33554432 match tcp host 192.168.240.50 host 1
cluster exec cap CCL int cluster buffer 33554432
```

Remarque : Ces tests ont été exécutés dans un environnement de travaux pratiques avec un trafic minimal dans le cluster. En production, essayez d'utiliser des filtres de capture aussi spécifiques que possible (par exemple, le port de destination et, dans la mesure du possible, le port source) pour minimiser le « bruit » dans les captures.

Étude de cas 1. Trafic symétrique (le propriétaire est également le directeur)

Observation 1. Les captures réinjecter-masquer affichent les paquets uniquement sur l'unité 1-1. Cela signifie que le flux dans les deux directions a transité par l'unité 1-1 (trafic symétrique) :

<#root>

```
firepower#
```

cluster exec show cap

```
unit-1-1(LOCAL):*******************************************************
capture CCL type raw-data interface cluster [Capturing - 33513 bytes]
capture CAPI type raw-data buffer 33554432 trace interface INSIDE [Buffer Full - 33553914 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO type raw-data buffer 33554432 trace interface OUTSIDE [Buffer Full - 33553914 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPI_RH type raw-data
```

reinject-hide

```
 buffer 33554432 interface INSIDE [Buffer Full -
```

33553914 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO_RH type raw-data
```

reinject-hide

```
 buffer 33554432 interface OUTSIDE [Buffer Full -
```

33553914 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80

unit-2-1:*******************************************************
capture CCL type raw-data interface cluster [Capturing - 23245 bytes]
capture CAPI type raw-data buffer 33554432 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO type raw-data buffer 33554432 trace interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPI_RH type raw-data
```

reinject-hide

```
 buffer 33554432 interface INSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO_RH type raw-data
reinject-hide
 buffer 33554432 interface OUTSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80

unit-3-1:**********************************************************
capture CCL type raw-data interface cluster [Capturing - 24815 bytes]
capture CAPI type raw-data buffer 33554432 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO type raw-data buffer 33554432 trace interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPI_RH type raw-data
reinject-hide
 buffer 33554432 interface INSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO_RH type raw-data
reinject-hide
 buffer 33554432 interface OUTSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
```

Observation 2. Analyse de l'indicateur de connexion pour le flux avec le port source 45954 :

```
<#root>
firepower#
cluster exec show conn


unit-1-1(LOCAL):*******************************************************
22 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 2 most enabled, 1 most in effect
```

```
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

45954

```
, idle 0:00:00, bytes 487413076,
```

flags UIO N1

```
unit-2-1:************************************************************
22 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 0 in use, 2 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 249 most enabled, 0 most in effect

unit-3-1:************************************************************
17 in use, 20 most used
Cluster:
fwd connections: 1 in use, 2 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:443 NP Identity Ifc 192.168.240.50:39698, idle 0:00:23, bytes 0, flags z
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

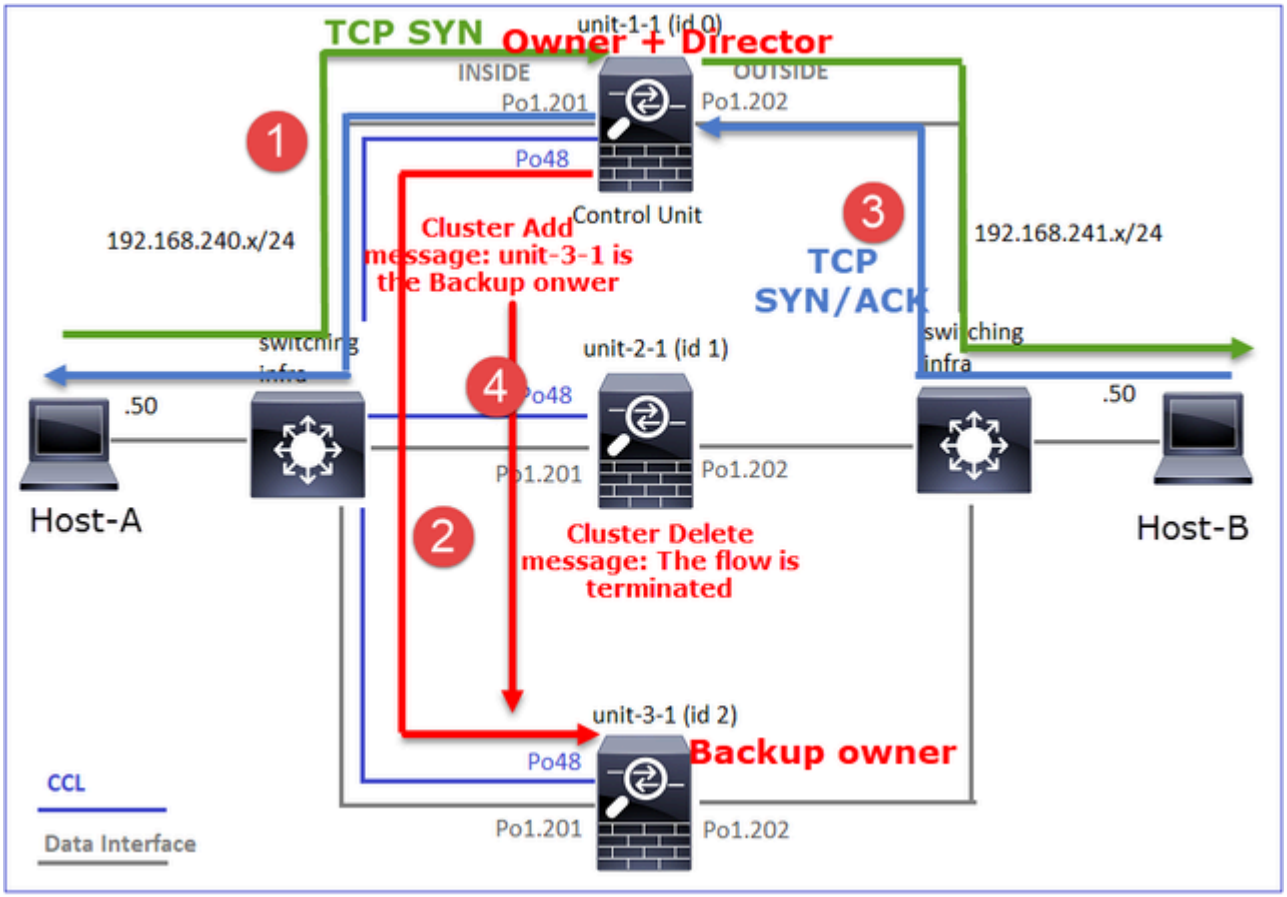
45954

```
, idle 0:00:06, bytes 0,
```

flags y

Unité	Drapeau	Remarque
Unité-1-1	UIO	· Flow Owner - L'unité gère le flux · Directeur - Puisque l'unité-3-1 a « y » et non « Y », cela signifie que l'unité-1-1 a été choisie comme directeur pour ce flux. Ainsi, puisqu'il est également propriétaire, une autre unité (unit-3-1 dans ce cas) a été élue comme propriétaire de secours
Unité-2-1	-	-
Unité-3-1	o	L'unité est propriétaire d'une sauvegarde

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 1-1. L'unité 1-1 devient le propriétaire du flux.
2. Unit-1-1 est également élu directeur de flux. Ainsi, il choisit également unit-3-1 comme propriétaire de sauvegarde (message d'ajout de cluster).
3. Le paquet TCP SYN/ACK arrive de l'hôte B à l'unité 3-1. Le flux est symétrique.
4. Une fois la connexion terminée, le propriétaire envoie un message de suppression de cluster pour supprimer les informations de flux du propriétaire de sauvegarde.

Observation 3. La capture avec trace montre que les deux directions passent uniquement par l'unité 1-1 :

Étape 1. Identifiez le flux et les paquets intéressants dans toutes les unités de cluster en fonction du port source :

<#root>

```
firepower#
```

```
cluster exec show capture CAPI | i 45954
```

```
unit-1-1(LOCAL):*******************************************************
1: 08:42:09.362697 802.1Q vlan#201 P0 192.168.240.50.45954 > 192.168.241.50.80: S 992089269:992089269(0
2: 08:42:09.363521 802.1Q vlan#201 P0 192.168.241.50.80 > 192.168.240.50.45954: S 4042762409:4042762409
3: 08:42:09.363827 802.1Q vlan#201 P0 192.168.240.50.45954 > 192.168.241.50.80: . ack 4042762410 win 22
…
unit-2-1:*******************************************************************

unit-3-1:*******************************************************************
```

<#root>

firepower#

cluster exec show capture CAPO | i 45954

```
unit-1-1(LOCAL):*******************************************************
1: 08:42:09.362987 802.1Q vlan#202 P0 192.168.240.50.45954 > 192.168.241.50.80: S 2732339016:2732339016
2: 08:42:09.363415 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45954: S 3603655982:3603655982
3: 08:42:09.363903 802.1Q vlan#202 P0 192.168.240.50.45954 > 192.168.241.50.80: . ack 3603655983 win 22
…
unit-2-1:*******************************************************************

unit-3-1:*******************************************************************
```

Étape 2. Puisqu'il s'agit d'un flux TCP, tracez les paquets d'échange en trois étapes. Comme on peut le voir dans cette sortie, unit-1-1 est le propriétaire. Pour des raisons de simplicité, les phases de trace non pertinentes sont omises :

<#root>

firepower#

show cap CAPI packet-number 1 trace

```
25985 packets captured
1: 08:42:09.362697 802.1Q vlan#201 P0 192.168.240.50.
```

45954

```
 > 192.168.241.50.80:
```

S

```
 992089269:992089269(0) win 29200 <mss 1460,sackOK,timestamp 495153655 0,nop,wscale 7>
...
Phase: 4
```

Type: CLUSTER-EVENT

```
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
```

Flow type: NO FLOW

I (0) got initial, attempting ownership.

Phase: 5

Type: CLUSTER-EVENT

Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (0) am becoming owner

...

Le trafic de retour (TCP SYN/ACK) :

<#root>

firepower#

show capture CAPO packet-number 2 trace

25985 packets captured
2: 08:42:09.363415 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45954:

S

3603655982:3603655982(0)

ack

2732339017 win 28960 <mss 1460,sackOK,timestamp 505509125 495153655,nop,wscale 7>
...
Phase: 3

Type: FLOW-LOOKUP

Subtype:
Result: ALLOW
Config:
Additional Information:

Found flow with id 9364, using existing flow

Observation 4. Les syslog du plan de données FTD indiquent la création et la fin de la connexion sur toutes les unités :

<#root>

```
firepower#

cluster exec show log | include 45954



unit-1-1

(LOCAL):*******************************************************
Dec 01 2020 08:42:09: %FTD-6-302013:

Built inbound TCP connection 9364

 for INSIDE:192.168.240.50/45954 (192.168.240.50/45954) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 08:42:18: %FTD-6-302014:

Teardown TCP connection 9364

 for INSIDE:192.168.240.50/45954 to OUTSIDE:192.168.241.50/80 duration 0:00:08 bytes 1024000440 TCP FIN

unit-2-1:*************************************************************



unit-3-1

:*************************************************************
Dec 01 2020 08:42:09: %FTD-6-302022:

Built backup stub TCP connection

 for INSIDE:192.168.240.50/45954 (192.168.240.50/45954) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 08:42:18: %FTD-6-302023:

Teardown backup TCP connection

 for INSIDE:192.168.240.50/45954 to OUTSIDE:192.168.241.50/80 duration 0:00:08 forwarded bytes 0 Cluste
```

Étude de cas 2. Trafic symétrique (propriétaire différent du directeur)

- Identique à l'étude de cas #1, mais dans cette étude de cas, le propriétaire d'un flux est une unité différente du directeur.
- Tous les résultats sont similaires à l'étude de cas #1. La principale différence par rapport à l'étude de cas #1 est l'indicateur « Y » qui remplace l'indicateur « y » du scénario 1.

Observation 1. Le propriétaire est différent du directeur.

Analyse de l'indicateur de connexion pour le flux avec le port source 46278 :

<#root>

```
firepower#

cluster exec show conn
```

```
unit-1-1(LOCAL):*******************************************************
23 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 2 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

46278

```
, idle 0:00:00, bytes 508848268, flags
```

UIO N1

```
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:46276, idle 0:00:03, bytes 0, flags aA N1

unit-2-1:*************************************************************
21 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 0 in use, 2 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

unit-3-1:*************************************************************
17 in use, 20 most used
Cluster:
fwd connections: 1 in use, 5 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 NP Identity Ifc 192.168.240.50:46276, idle 0:00:02, bytes 0, flags z
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

46278

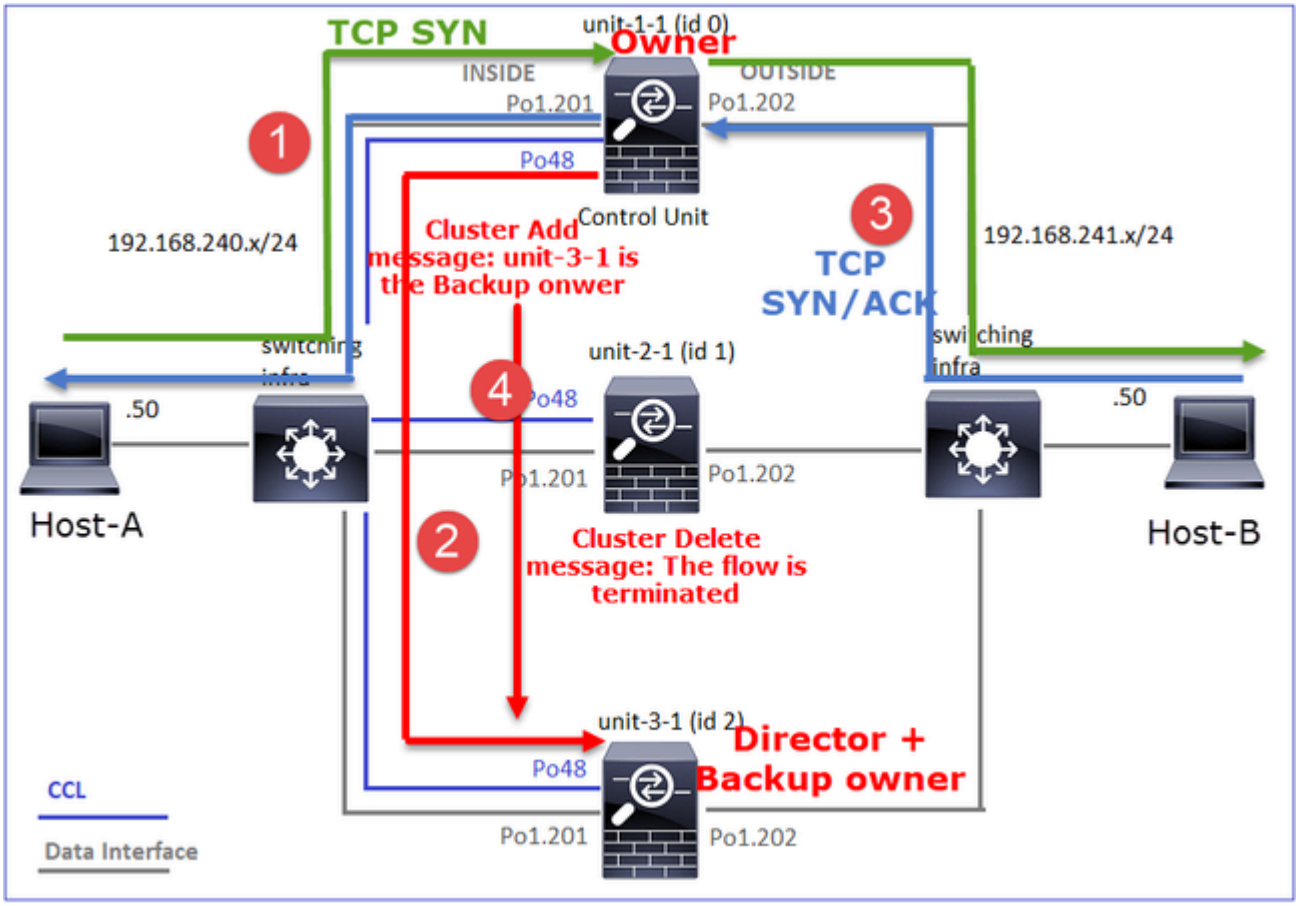
```
, idle 0:00:06, bytes 0,
```

flags Y

Unité	Drapeau	Remarque
Unité-1-1	UIO	· Flow Owner - L'unité gère le flux
Unité-2-1	-	-

Unité-3-1	O	· Directeur et propriétaire de la sauvegarde - L'unité 3-1 porte le drapeau Y (Directeur).

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 1-1. L'unité 1-1 devient le propriétaire du flux.
2. Unit-3-1 est élu directeur de flux. Unit-3-1 est également le propriétaire de la sauvegarde (message « cluster add » sur UDP 4193 sur la CCL).
3. Le paquet TCP SYN/ACK arrive de l'hôte B à l'unité 3-1. Le flux est symétrique.
4. Une fois la connexion terminée, le propriétaire envoie sur la CCL un message « cluster delete » sur UDP 4193 pour supprimer les informations de flux du propriétaire de sauvegarde.

Observation 2. La capture avec trace montre que les deux directions passent uniquement par l'unité 1-1.

Étape 1. Utilisez la même approche que dans l'étude de cas 1 pour identifier le flux et les paquets d'intérêt dans toutes les unités de cluster en fonction du port source :

<#root>

```
firepower#
```

```
cluster exec show cap CAPI | include 46278
```

```
unit-1-1

(LOCAL):*******************************************************
3: 11:01:44.841631 802.1Q vlan#201 P0 192.168.240.50.46278 > 192.168.241.50.80:

S

 1972783998:1972783998(0) win 29200 <mss 1460,sackOK,timestamp 503529072 0,nop,wscale 7>
4: 11:01:44.842317 802.1Q vlan#201 P0 192.168.241.50.80 > 192.168.240.50.46278:

S

 3524167695:3524167695(0)

ack

 1972783999 win 28960 <mss 1380,sackOK,timestamp 513884542 503529072,nop,wscale 7>
5: 11:01:44.842592 802.1Q vlan#201 P0 192.168.240.50.46278 > 192.168.241.50.80: . ack 3524167696 win 22
…
unit-2-1:*******************************************************

unit-3-1:*******************************************************
firepower#
```

Capture sur l'interface OUTSIDE :

<#root>

```
firepower#

cluster exec show cap CAPO | include 46278


unit-1-1

(LOCAL):*******************************************************
3: 11:01:44.841921 802.1Q vlan#202 P0 192.168.240.50.46278 > 192.168.241.50.80:

S

 2153055699:2153055699(0) win 29200 <mss 1380,sackOK,timestamp 503529072 0,nop,wscale 7>
4: 11:01:44.842226 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.46278:

S

 3382481337:3382481337(0)

ack

 2153055700 win 28960 <mss 1460,sackOK,timestamp 513884542 503529072,nop,wscale 7>
5: 11:01:44.842638 802.1Q vlan#202 P0 192.168.240.50.46278 > 192.168.241.50.80: . ack 3382481338 win 22

unit-2-1:*******************************************************

unit-3-1:*******************************************************
firepower#
```

Étape 2. Concentration sur les paquets entrants (TCP SYN et TCP SYN/ACK) :

<#root>

```
firepower#
cluster exec show cap CAPI packet-number 3 trace

unit-1-1(LOCAL):*******************************************************

824 packets captured

3: 11:01:44.841631 802.1Q vlan#201 P0 192.168.240.50.46278 > 192.168.241.50.80:
S
 1972783998:1972783998(0) win 29200 <mss 1460,sackOK,timestamp 503529072 0,nop,wscale 7>
…

Phase: 4

Type: CLUSTER-EVENT

Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (0) got initial, attempting ownership.


Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (0) am becoming owner
```

Suivez le SYN/ACK sur l'unité 1-1 :

<#root>

```
firepower#
cluster exec show cap CAPO packet-number 4 trace
```

```
unit-1-1(LOCAL):*******************************************************

4: 11:01:44.842226 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.
```

46278

```
:
```

S

```
 3382481337:3382481337(0)
```

ack

```
 2153055700 win 28960 <mss 1460,sackOK,timestamp 513884542 503529072,nop,wscale 7>
Phase: 3
Type: FLOW-LOOKUP
Subtype:
Result: ALLOW
Config:
Additional Information:
```

Found flow with id 9583, using existing flow

Observation 3. Les syslogs du plan de données FTD montrent la création et la fin de la connexion sur le propriétaire et le propriétaire de sauvegarde :

<#root>

```
firepower#
```

cluster exec show log | include 46278

```
unit-1-1(LOCAL):*******************************************************
Dec 01 2020 11:01:44: %FTD-6-302013:
```

Built inbound TCP connection

```
 9583 for INSIDE:192.168.240.50/46278 (192.168.240.50/46278) to OUTSIDE:192.168.241.50/80 (192.168.241.
Dec 01 2020 11:01:53: %FTD-6-302014:
```

Teardown TCP connection

```
 9583 for INSIDE:192.168.240.50/46278 to OUTSIDE:192.168.241.50/80 duration 0:00:08 bytes 1024001808 TC

unit-2-1:*************************************************************

unit-3-1:*************************************************************
Dec 01 2020 11:01:44: %FTD-6-302022:
```

Built director stub TCP connection

```
 for INSIDE:192.168.240.50/46278 (192.168.240.50/46278) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 11:01:53: %FTD-6-302023:
```

Teardown director TCP connection

```
 for INSIDE:192.168.240.50/46278 to OUTSIDE:192.168.241.50/80 duration 0:00:08 forwarded bytes 0 Cluste
```

Étude de cas 3. Trafic asymétrique (le directeur achemine le trafic)

Observation 1. Les captures réinjecter-masquer montrent les paquets sur l'unité 1-1 et l'unité 2-1 (flux asymétrique) :

<#root>

```
firepower#

cluster exec show cap


unit-1-1(LOCAL):*******************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554320 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Buffer Full - 98552 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 98552 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data

reinject-hide

 buffer 100000 interface

INSIDE

 [Buffer Full -

98552 bytes

]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data

reinject-hide

 buffer 100000 interface

OUTSIDE

 [Buffer Full -

99932 bytes

]
match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-2-1:*************************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553268 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99052 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data

reinject-hide

 buffer 100000 interface

OUTSIDE
```

```
 [Buffer Full -

99052 bytes

]
match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:*************************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 53815 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Capturing - 658 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data reinject-hide buffer 100000 interface OUTSIDE [Capturing - 658 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

Observation 2. Analyse de l'indicateur de connexion pour le flux avec le port source 46502.

<#root>

```
firepower#

cluster exec show conn



unit-1-1

(LOCAL):*****************************************************
23 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 2 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:

46502

, idle 0:00:00, bytes 448760236,

flags UIO N1


TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:46500, idle 0:00:06, bytes 0, flags aA N1



unit-2-1

:************************************************************
21 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 1 in use, 2 most used
```

```
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

46502

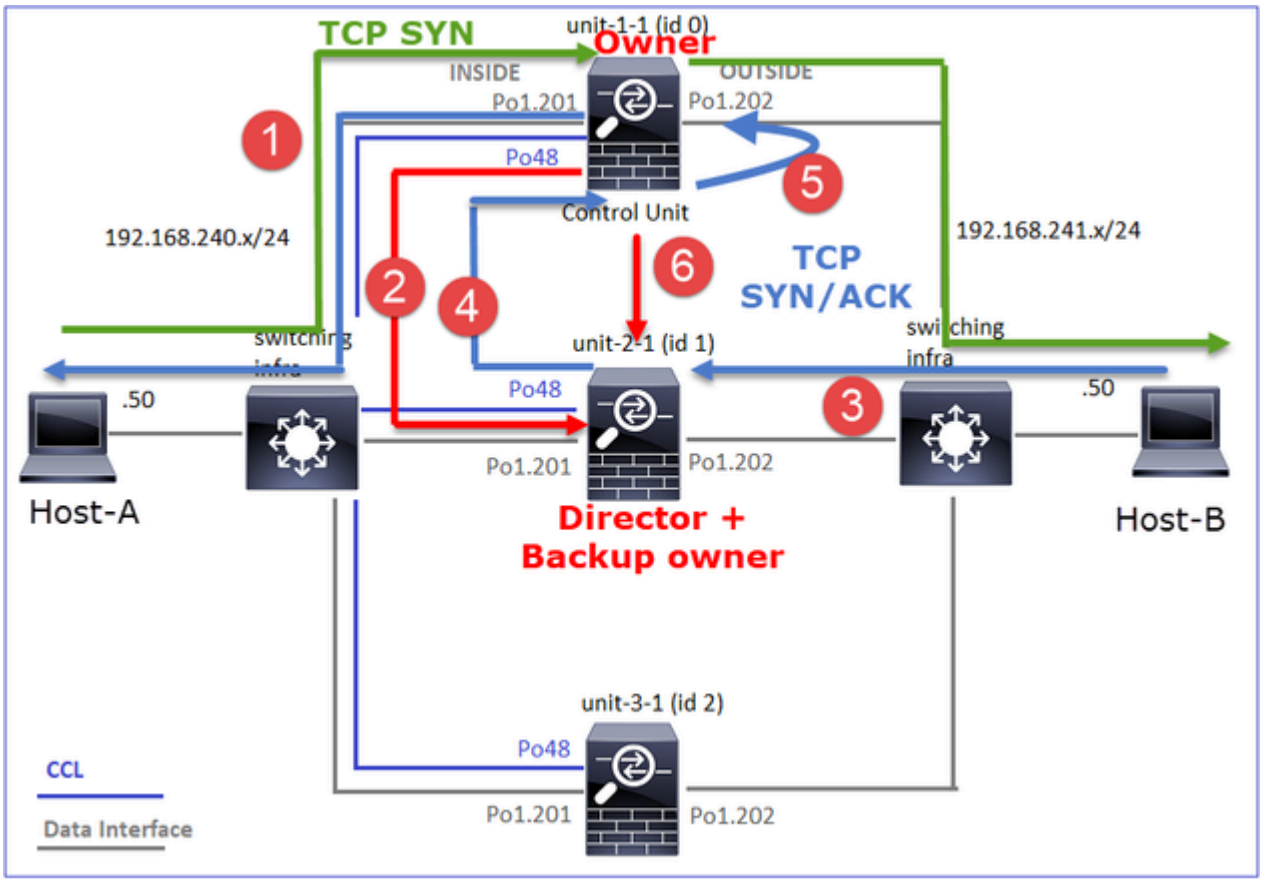
```
, idle 0:00:00, bytes 0,
```

flags Y

```
unit-3-1:**************************************************************
17 in use, 20 most used
Cluster:
fwd connections: 1 in use, 5 most used
dir connections: 0 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect
```

Unité	Drapeau	Remarque
Unité-1-1	UIO	· Flow Owner : l'unité gère le flux.
Unité-2-1	O	· Directeur - Comme l'unité 2-1 a la mention « Y », cela signifie que l'unité 2-1 a été choisie comme directeur pour ce flux. · Propriétaire de sauvegarde · Enfin, bien que cela ne soit pas évident à partir de ce résultat, les résultats des commandes show capture et show log indiquent clairement que l'unité 2-1 transmet ce flux au propriétaire (bien que techniquement, il ne soit pas considéré comme un transmetteur dans ce scénario). Remarque : Une unité ne peut pas être à la fois directeur (flux Y) et redirecteur (flux z), ces 2 rôles s'excluent mutuellement. Les directeurs (flux Y) peuvent toujours transférer le trafic. Reportez-vous au résultat de la commande show log plus loin dans cette étude de cas.
Unité-3-1	-	-

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 1-1. L'unité 1-1 devient le propriétaire du flux.
2. Unit-2-1 est élu directeur de flux et propriétaire de sauvegarde. Le propriétaire du flux envoie un message de monodiffusion « cluster add » sur UDP 4193 pour informer le propriétaire de sauvegarde du flux.
3. Le paquet TCP SYN/ACK arrive de l'hôte B à l'unité 2-1. Le flux est asymétrique.
4. Unit-2-1 transfère le paquet via la CCL au propriétaire (en raison du cookie TCP SYN).
5. Le propriétaire réinjecte le paquet sur l'interface OUTSIDE, puis le transfère vers l'hôte A.
6. Une fois la connexion terminée, le propriétaire envoie un message de suppression de cluster pour supprimer les informations de flux du propriétaire de sauvegarde.

Observation 3. La capture avec trace montre le trafic asymétrique et la redirection de l'unité 2-1 vers l'unité 1-1.

Étape 1 : identification des paquets appartenant au flux d'intérêt (port 46502)

<#root>

```
firepower#
```

```
cluster exec show capture CAPI | include 46502
```

```
unit-1-1(LOCAL):*******************************************************
3: 12:58:33.356121 802.1Q vlan#201 P0 192.168.240.50.46502 > 192.168.241.50.80: S 4124514680:4124514680
4: 12:58:33.357037 802.1Q vlan#201 P0 192.168.241.50.80 > 192.168.240.50.46502: S 883000451:883000451(0
5: 12:58:33.357357 802.1Q vlan#201 P0 192.168.240.50.46502 > 192.168.241.50.80: . ack 883000452 win 229
unit-2-1:*******************************************************

unit-3-1:*******************************************************
```

La direction de retour :

<#root>

```
firepower#
```

cluster exec show capture CAPO | include 46502

```
unit-1-1(LOCAL):*******************************************************
3: 12:58:33.356426 802.1Q vlan#202 P0 192.168.240.50.46502 > 192.168.241.50.80: S 1434968587:1434968587
4: 12:58:33.356915 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.46502: S 4257314722:4257314722
5: 12:58:33.357403 802.1Q vlan#202 P0 192.168.240.50.46502 > 192.168.241.50.80: . ack 4257314723 win 22

unit-2-1:*******************************************************************
1: 12:58:33.359249 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.46502: S 4257314722:4257314722
2: 12:58:33.360302 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.46502: . ack 1434968736 win 23
3: 12:58:33.361004 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.46502: . 4257314723:4257316091
…

unit-3-1:*******************************************************************
```

Étape 2 : suivi des paquets Par défaut, seuls les 50 premiers paquets entrants sont suivis. Pour des raisons de simplicité, les phases de trace non pertinentes sont omises.

Unit-1-1 (propriétaire)

<#root>

```
firepower#
```

cluster exec show capture CAPI packet-number 3 trace

```
unit-1-1(LOCAL):*******************************************************
3: 12:58:33.356121 802.1Q vlan#201 P0 192.168.240.50.
```

46502

```
 > 192.168.241.50.80:
```

S

```
 4124514680:4124514680(0) win 29200 <mss 1460,sackOK,timestamp 510537534 0,nop,wscale 7>
...
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW
```

I (0) got initial, attempting ownership.

```
Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW
```

I (0) am becoming owner

Unité-2-1 (redirecteur)

Le trafic de retour (TCP SYN/ACK) L'unité intéressante est l'unité 2-1, qui est le directeur/propriétaire de sauvegarde et transfère le trafic au propriétaire :

<#root>

```
firepower#
```

cluster exec unit unit-2-1 show capture CAPO packet-number 1 trace

```
1: 12:58:33.359249 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.
```

46502

```
: S 4257314722:4257314722(0) ack 1434968588 win 28960 <mss 1460,sackOK,timestamp 520893004 510537534,no
...
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (1) got initial, attempting ownership.

```
Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (1) am early redirecting to (0) due to matching action (-1).

Observation 4. Les syslog du plan de données FTD indiquent la création et la fin de la connexion sur toutes les unités :

<#root>

```
firepower#

cluster exec show log | i 46502


unit-1-1(LOCAL):*******************************************************
Dec 01 2020 12:58:33: %FTD-6-302013:

B

uilt inbound TCP connection

 9742 for INSIDE:192.168.240.50/46502 (192.168.240.50/46502) to OUTSIDE:192.168.241.50/80 (192.168.241.
Dec 01 2020 12:59:02: %FTD-6-302014:

Teardown TCP connection

 9742 for INSIDE:192.168.240.50/46502 to OUTSIDE:192.168.241.50/80 duration 0:00:28 bytes 2048000440 TC

unit-2-1:*******************************************************************
Dec 01 2020 12:58:33: %FTD-6-302022:

Built forwarder stub TCP connection

 for OUTSIDE:192.168.241.50/80 (192.168.241.50/80) to unknown:192.168.240.50/46502 (192.168.240.50/4650
Dec 01 2020 12:58:33: %FTD-6-302023:

Teardown forwarder TCP connection

 for OUTSIDE:192.168.241.50/80 to unknown:192.168.240.50/46502 duration 0:00:00 forwarded bytes 0 Forwa
Dec 01 2020 12:58:33: %FTD-6-302022:

Built director stub TCP connection

 for INSIDE:192.168.240.50/46502 (192.168.240.50/46502) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80
Dec 01 2020 12:59:02: %FTD-6-302023:

Teardown director TCP connection

 for INSIDE:192.168.240.50/46502 to OUTSIDE:192.168.241.50/80 duration 0:00:28 forwarded bytes 20483163

unit-3-1:*******************************************************************
firepower#
```

Étude de cas 4. Trafic asymétrique (le propriétaire est le directeur)

Observation 1. Les captures réinjecter-masquer montrent les paquets sur l'unité 1-1 et l'unité 2-1 (flux asymétrique) :

<#root>

```
firepower#

cluster exec show cap
```

```
unit-1-1(LOCAL):*******************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554229 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Buffer Full - 98974 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 98974 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data
```

reinject-hide

```
 buffer 100000 interface
```

INSIDE

```
 [Buffer Full -
```

98974 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data
```

reinject-hide

```
 buffer 100000 interface
```

OUTSIDE

```
 [Buffer Full -
```

99924 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-2-1:*************************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33552925 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99052 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data
```

reinject-hide

```
 buffer 100000 interface OUTSIDE [Buffer Full -
```

99052 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:*************************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 227690 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Capturing - 4754 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data reinject-hide buffer 100000 interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

Observation 2. Analyse de l’indicateur de connexion pour le flux avec le port source 46916.

<#root>

```
firepower#
 cluster exec show conn


unit-1-1
(LOCAL):*******************************************************
23 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
46916
, idle 0:00:00, bytes 414682616,
flags UIO N1


unit-2-1
:*************************************************************
21 in use, 271 most used
Cluster:
fwd connections: 1 in use, 2 most used
dir connections: 0 in use, 2 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 NP Identity Ifc 192.168.240.50:
46916
, idle 0:00:00, bytes 0,
flags z


unit-3-1
:*************************************************************
17 in use, 20 most used
Cluster:
```

```
fwd connections: 0 in use, 5 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

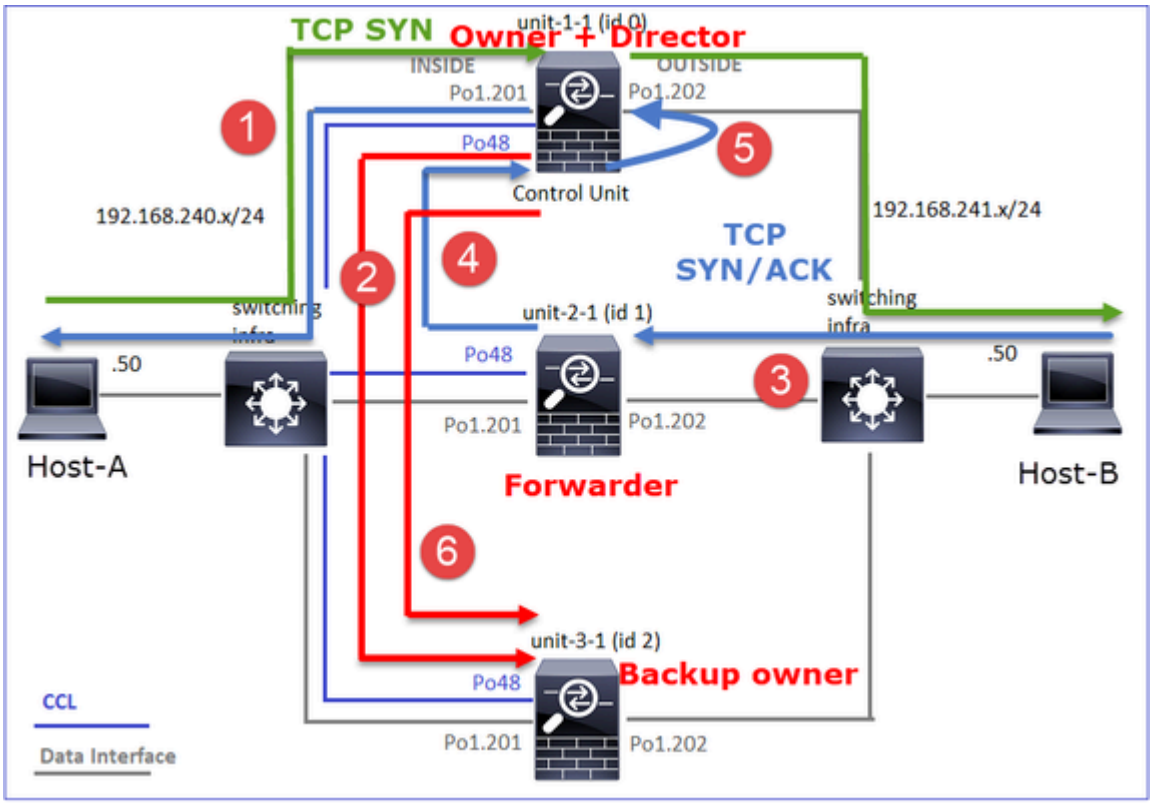
46916

```
, idle 0:00:04, bytes 0,
```

flags y

Unité	Drapeau	Remarque
Unité-1-1	UIO	· Flow Owner - L'unité gère le flux · Directeur - Puisque l'unité-3-1 a « y » et non « Y », cela signifie que l'unité-1-1 a été choisie comme directeur pour ce flux. Ainsi, puisqu'il est également propriétaire, une autre unité (unit-3-1 dans ce cas) a été élue comme propriétaire de secours
Unité-2-1	z	· Transporteur
Unité-3-1	o	- Propriétaire de la sauvegarde

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 1-1. L'unité 1-1 devient le propriétaire du flux et est élue comme directeur.
2. L'unité 3-1 est sélectionnée comme propriétaire de sauvegarde. Le propriétaire du flux envoie un message « cluster add » de monodiffusion sur UDP 4193 pour informer le propriétaire de sauvegarde du flux.
3. Le paquet TCP SYN/ACK arrive de l'hôte B à l'unité 2-1. Le flux est asymétrique.
4. Unit-2-1 transfère le paquet via la CCL au propriétaire (en raison du cookie TCP SYN).
5. Le propriétaire réinjecte le paquet sur l'interface OUTSIDE, puis le transfère vers l'hôte A.
6. Une fois la connexion terminée, le propriétaire envoie un message de suppression de cluster pour supprimer les informations de flux du propriétaire de sauvegarde.

Observation 3. La capture avec trace montre le trafic asymétrique et la redirection de l'unité 2-1 vers l'unité 1-1.

Unit-2-1 (redirecteur) :

<#root>

```
firepower#
cluster exec unit unit-2-1 show capture CAPO packet-number 1 trace

1: 16:11:33.653164 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.
46916
:
S
 1331019196:1331019196(0)
```

ack

```
 3089755618 win 28960 <mss 1460,sackOK,timestamp 532473211 522117741,nop,wscale 7>
...
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (1) got initial, attempting ownership.

```
Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (1) am early redirecting to (0) due to matching action (-1).

Observation 4. Les syslog du plan de données FTD indiquent la création et la fin de la connexion sur toutes les unités :

- Unit-1-1 (propriétaire)
- Unité-2-1 (redirecteur)
- Unit-3-1 (propriétaire de la sauvegarde)

<#root>

```
firepower#
```

cluster exec show log | i 46916

```
unit-1-1(LOCAL):*******************************************************
Dec 01 2020 16:11:33: %FTD-6-302013:
```

Built inbound TCP connection

```
 10023 for INSIDE:192.168.240.50/46916 (192.168.240.50/46916) to OUTSIDE:192.168.241.50/80 (192.168.241
Dec 01 2020 16:11:42: %FTD-6-302014:
```

Teardown TCP connection

```
 10023 for INSIDE:192.168.240.50/46916 to OUTSIDE:192.168.241.50/80 duration 0:00:09 bytes 1024010016 T

unit-2-1:*************************************************************
Dec 01 2020 16:11:33: %FTD-6-302022:
```

Built forwarder stub TCP connection

```
 for OUTSIDE:192.168.241.50/80 (192.168.241.50/80) to unknown:192.168.240.50/46916 (192.168.240.50/4691
Dec 01 2020 16:11:42: %FTD-6-302023:

Teardown forwarder TCP connection

 for OUTSIDE:192.168.241.50/80 to unknown:192.168.240.50/46916 duration 0:00:09 forwarded bytes 1024009

unit-3-1:************************************************************
Dec 01 2020 16:11:33: %FTD-6-302022:

Built backup stub TCP connection

 for INSIDE:192.168.240.50/46916 (192.168.240.50/46916) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80
Dec 01 2020 16:11:42: %FTD-6-302023:

Teardown backup TCP connection

 for INSIDE:192.168.240.50/46916 to OUTSIDE:192.168.241.50/80 duration 0:00:09 forwarded bytes 0 Cluste
```

Étude de cas 5. Trafic asymétrique (le propriétaire est différent du directeur)

Observation 1. Les captures réinjecter-masquer montrent les paquets sur l'unité 1-1 et l'unité 2-1 (flux asymétrique) :

<#root>

```
firepower#

cluster exec show cap



unit-1-1

(LOCAL):*****************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553207 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Buffer Full - 99396 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99224 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data

reinject-hide

 buffer 100000 interface

INSIDE

 [Buffer Full -

99396 bytes

]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data

reinject-hid

e buffer 100000 interface

OUTSIDE
```

```
 [Buffer Full -
```

99928 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

unit-2-1

```
:************************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554251 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99052 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data
```

reinject-hide

```
 buffer 100000 interface
```

OUTSIDE

```
 [Buffer Full -
```

99052 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:************************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 131925 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Capturing - 2592 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data reinject-hide buffer 100000 interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

Observation 2. Analyse de l’indicateur de connexion pour le flux avec le port source 4694 :

<#root>

```
firepower#
```

cluster exec show conn

unit-1-1

```
(LOCAL):*************************************************
23 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
```

```
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

46994

```
, idle 0:00:00, bytes 406028640,
```

flags UIO N1

unit-2-1

```
:************************************************************
22 in use, 271 most used
Cluster:
fwd connections: 1 in use, 2 most used
dir connections: 0 in use, 2 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 NP Identity Ifc 192.168.240.50:
```

46994

```
, idle 0:00:00, bytes 0,
```

flags z

unit-3-1

```
:************************************************************
17 in use, 20 most used
Cluster:
fwd connections: 2 in use, 5 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

46994

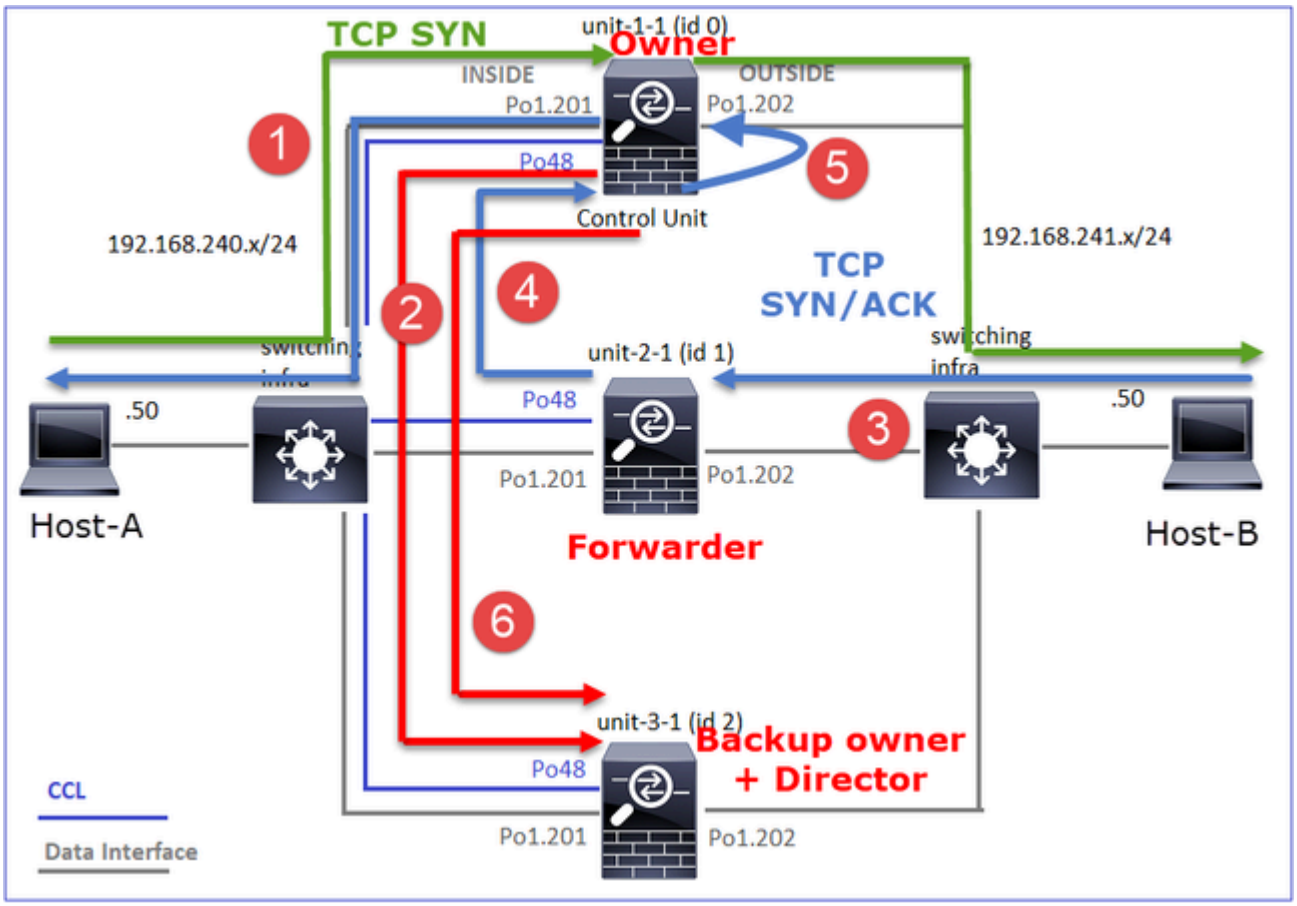
```
, idle 0:00:05, bytes 0,
```

flags Y

Unité	Drapeau	Remarque

Unité-1-1	UIO	· Flow Owner - L'unité gère le flux
Unité-2-1	z	· Transporteur
Unité-3-1	O	· Propriétaire de sauvegarde ·Direction

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 1-1. L'unité 1-1 devient le propriétaire du flux.
2. Unit-3-1 est élu comme directeur et propriétaire de sauvegarde. Le propriétaire du flux envoie un message de monodiffusion « cluster add » sur UDP 4193 pour informer le propriétaire de sauvegarde du flux.
3. Le paquet TCP SYN/ACK arrive de l'hôte B vers l'unité 2-1. Le flux est asymétrique
4. Unit-2-1 transfère le paquet via la CCL au propriétaire (en raison du cookie TCP SYN).
5. Le propriétaire réinjecte le paquet sur l'interface OUTSIDE, puis le transfère vers l'hôte A.
6. Une fois la connexion terminée, le propriétaire envoie un message de suppression de cluster pour supprimer les informations de flux du propriétaire de sauvegarde.

Observation 3. La capture avec trace montre le trafic asymétrique et la redirection de l'unité 2-1 vers l'unité 1-1.

Unit-1-1 (propriétaire) :

<#root>

```
firepower#
```

cluster exec show cap CAPI packet-number 1 trace

```
unit-1-1(LOCAL):******************************************************
…
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW
```

I (0) got initial, attempting ownership.

```
Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW
```

I (0) am becoming owner

Unit-2-1 (redirecteur) :

<#root>

```
firepower#
```

cluster exec unit unit-2-1 show cap CAPO packet-number 1 trace

```
1: 16:46:44.232074 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.
```

46994

```
: S 2863659376:2863659376(0) ack 2879616990 win 28960 <mss 1460,sackOK,timestamp 534583774 524228304,no
…
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
```

```
Flow type: NO FLOW
```

I (1) got initial, attempting ownership.

```
Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (1) am early redirecting to (0) due to matching action (-1).

Observation 4. Les syslog du plan de données FTD indiquent la création et la fin de la connexion sur toutes les unités :

- Unit-1-1 (propriétaire)
- Unité-2-1 (redirecteur)
- Unit-3-1 (propriétaire/directeur de sauvegarde)

<#root>

```
firepower#
```

cluster exec show log | i 46994

```
unit-1-1(LOCAL):*******************************************************
Dec 01 2020 16:46:44: %FTD-6-302013:
```

Built inbound TCP connection

```
 10080 for INSIDE:192.168.240.50/46994 (192.168.240.50/46994) to OUTSIDE:192.168.241.50/80 (192.168.241
Dec 01 2020 16:46:53: %FTD-6-302014:
```

Teardown TCP connection

```
 10080 for INSIDE:192.168.240.50/46994 to OUTSIDE:192.168.241.50/80 duration 0:00:09 bytes 1024000440 T

unit-2-1:*******************************************************************
Dec 01 2020 16:46:44: %FTD-6-302022:
```

Built forwarder stub TCP connection

```
 for OUTSIDE:192.168.241.50/80 (192.168.241.50/80) to unknown:192.168.240.50/46994 (192.168.240.50/4699
Dec 01 2020 16:46:53: %FTD-6-302023:
```

Teardown forwarder TCP connection

```
 for OUTSIDE:192.168.241.50/80 to unknown:192.168.240.50/46994 duration 0:00:09 forwarded bytes 1024000

unit-3-1:*******************************************************************
Dec 01 2020 16:46:44: %FTD-6-302022:
```

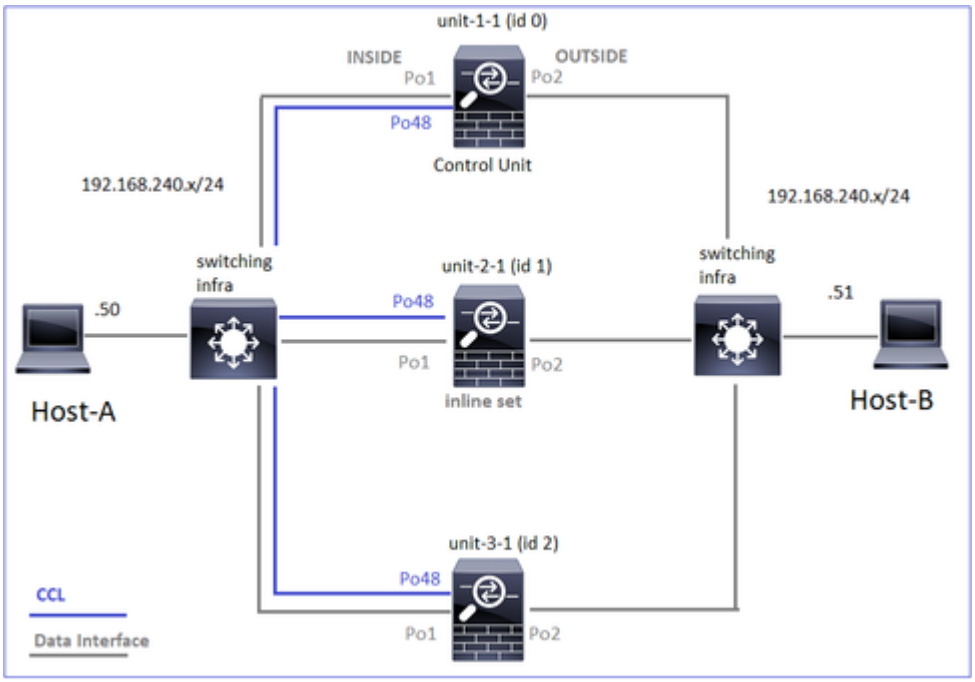
Built director stub TCP connection

```
 for INSIDE:192.168.240.50/46994 (192.168.240.50/46994) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 16:46:53: %FTD-6-302023:
```

Teardown director TCP connection

```
 for INSIDE:192.168.240.50/46994 to OUTSIDE:192.168.241.50/80 duration 0:00:09 forwarded bytes 0 Cluste
```

Pour les études de cas suivantes, la topologie utilisée est basée sur un cluster avec des ensembles en ligne :



Étude de cas 6. Trafic asymétrique (en ligne, le propriétaire est le directeur)

Observation 1. Les captures réinjecter-masquer montrent les paquets sur l'unité 1-1 et l'unité 2-1 (flux asymétrique). En outre, le propriétaire est unit-2-1 (il y a des paquets sur les deux, INSIDE et OUTSIDE i. interfaces pour les captures réinjecter-masquer, tandis que l'unité 1-1 a uniquement sur EXTERNE) :

<#root>

```
firepower#
```

cluster exec show cap

unit-1-1

```
(LOCAL):*******************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553253 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523432 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data
```

reinject-hide

 interface

OUTSIDE

 [Buffer Full -

523432 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www

unit-2-1

:**
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554312 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523782 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Buffer Full - 523782 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data

reinject-hide

 interface

OUTSIDE

 [Buffer Full -

524218 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data

reinject-hide

 interface

INSIDE

 [Buffer Full -

523782 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

unit-3-1:**
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 53118 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data reinject-hide interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www

Observation 2. Analyse de l’indicateur de connexion pour le flux avec le port source 51844.

```
<#root>

firepower#

cluster exec show conn addr 192.168.240.51



unit-1-1

(LOCAL):*******************************************************
30 in use, 102 most used
Cluster:
fwd connections: 1 in use, 1 most used
dir connections: 2 in use, 122 most used
centralized connections: 3 in use, 39 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.240.51:80 NP Identity Ifc 192.168.240.50:

51844

, idle 0:00:00, bytes 0,

flags z



unit-2-1

:*************************************************************
23 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 4 in use, 26 most used
centralized connections: 0 in use, 14 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:

51844

, idle 0:00:00, bytes 231214400,

flags b N



unit-3-1

:*************************************************************
20 in use, 55 most used
Cluster:
fwd connections: 0 in use, 5 most used
dir connections: 1 in use, 127 most used
```

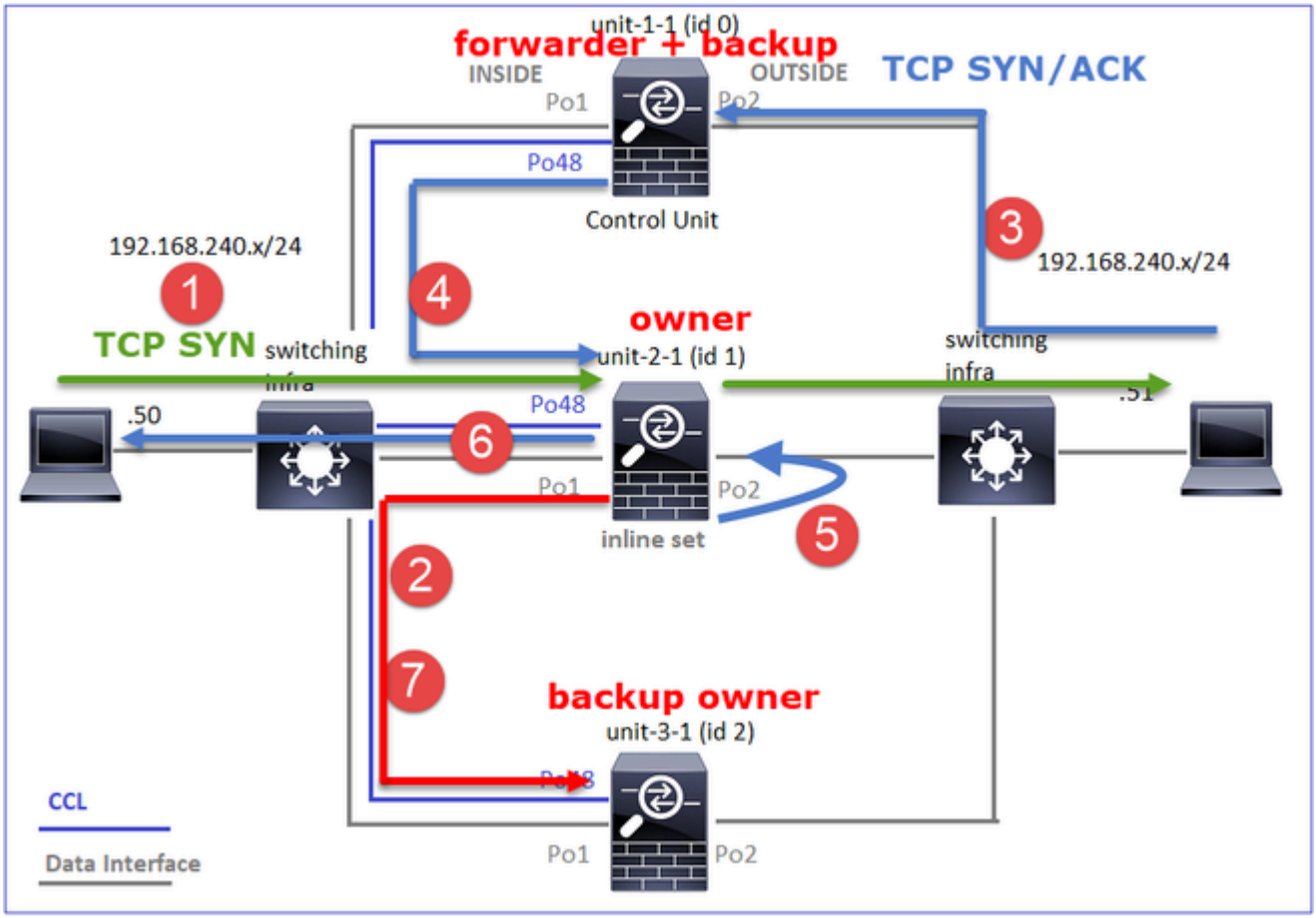
```
centralized connections: 0 in use, 24 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:51844, idle 0:00:01, bytes 0,
```

flags y

Unité	Drapeau	Remarque
Unité-1-1	z	· Transporteur
Unité-2-1	b N	· Flow Owner - L'unité gère le flux
Unité-3-1	o	· Propriétaire de sauvegarde

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 2-1. L'unité 2-1 devient le propriétaire du flux et est élue comme directeur.
2. Unit-3-1 est élu propriétaire de la sauvegarde. Le propriétaire du flux envoie un message de

monodiffusion « cluster add » sur UDP 4193 pour informer le propriétaire de sauvegarde du flux.
3. Le paquet TCP SYN/ACK arrive de l'hôte B à l'unité 1-1. Le flux est asymétrique.
4. Unit-1-1 transmet le paquet via la CCL au directeur (unit-2-1).
5. Unit-2-1 est également le propriétaire et réinjecte le paquet sur l'interface OUTSIDE.
6. L'unité 2-1 transfère le paquet vers l'hôte A.
7. Une fois la connexion terminée, le propriétaire envoie un message de suppression de cluster pour supprimer les informations de flux du propriétaire de sauvegarde.

Observation 3. La capture avec trace montre le trafic asymétrique et la redirection de l'unité 1-1 vers l'unité 2-1.

Unité-2-1 (propriétaire/directeur) :

<#root>

```
firepower#
cluster exec unit unit-2-1 show cap CAPI packet-number 1 trace


1: 18:10:12.842912 192.168.240.50.51844 > 192.168.240.51.80:
S
 4082593463:4082593463(0) win 29200 <mss 1460,sackOK,timestamp 76258053 0,nop,wscale 7>
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW


I (1) got initial, attempting ownership.



Phase: 2
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW


I (1) am becoming owner
```

Unité-1-1 (redirecteur) :

<#root>

```
firepower#
```

cluster exec show cap CAPO packet-number 1 trace

```
unit-1-1(LOCAL):*****************************************************

1: 18:10:12.842317 192.168.240.51.80 > 192.168.240.50.51844: S 2339579109:2339579109(0) ack 4082593464 w
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (0) am asking director (1).

Trafic de retour (TCP SYN/ACK)

Unité-2-1 (propriétaire/directeur) :

<#root>

```
firepower#
```

cluster exec unit unit-2-1 show cap CAPO packet-number 2 trace

```
2: 18:10:12.843660 192.168.240.51.80 > 192.168.240.50.51844: S 2339579109:2339579109(0) ack 4082593464 w
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: FULL
```

I (1) am owner, update sender (0).

```
Phase: 2
Type: FLOW-LOOKUP
Subtype:
Result: ALLOW
Config:
Additional Information:
```

Found flow with id 7109, using existing flow

Observation 4. Les syslog du plan de données FTD indiquent la création et la fin de la connexion sur toutes les unités :

- Unit-1-1 (propriétaire)
- Unité-2-1 (redirecteur)
- Unit-3-1 (propriétaire/directeur de sauvegarde)

<#root>

```
firepower#

cluster exec show log | include 51844


unit-1-1(LOCAL):*******************************************************
Dec 02 2020 18:10:12: %FTD-6-302022:

Built forwarder stub TCP connection

 for OUTSIDE:192.168.240.51/80 (192.168.240.51/80) to unknown:192.168.240.50/51844 (192.168.240.50/5184
Dec 02 2020 18:10:22: %FTD-6-302023:

Teardown forwarder TCP connection

 for OUTSIDE:192.168.240.51/80 to unknown:192.168.240.50/51844 duration 0:00:09 forwarded bytes 1024001

unit-2-1:*************************************************************
Dec 02 2020 18:10:12: %FTD-6-302303:

Built TCP state-bypass connection

 7109 from INSIDE:192.168.240.50/51844 (192.168.240.50/51844) to OUTSIDE:192.168.240.51/80 (192.168.240
Dec 02 2020 18:10:22: %FTD-6-302304:

Teardown TCP state-bypass connection

 7109 from INSIDE:192.168.240.50/51844 to OUTSIDE:192.168.240.51/80 duration 0:00:09 bytes 1024001888 T

unit-3-1:*************************************************************
Dec 02 2020 18:10:12: %FTD-6-302022:

Built backup stub TCP connection

 for INSIDE:192.168.240.50/51844 (192.168.240.50/51844) to OUTSIDE:192.168.240.51/80 (192.168.240.51/80
Dec 02 2020 18:10:22: %FTD-6-302023:

Teardown backup TCP connection

 for INSIDE:192.168.240.50/51844 to OUTSIDE:192.168.240.51/80 duration 0:00:09 forwarded bytes 0 Cluste
```

Étude de cas 7. Trafic asymétrique (en ligne, le propriétaire est différent du directeur)

Le propriétaire est l'unité 2-1 (il y a des paquets sur les deux interfaces, INSIDE et OUTSIDE pour les captures réinjecter-masquer, alors que l'unité 3-1 n'a que sur OUTSIDE) :

<#root>

```
firepower#
cluster exec show cap

unit-1-1(LOCAL):*******************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 13902 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Capturing - 90 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data reinject-hide interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www



unit-2-1

:*********************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553936 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523126 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Buffer Full - 523126 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data

reinject-hid

e

 interface

OUTSIDE

 [Buffer Full -

524230 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data

reinject-hide

 interface

INSIDE

 [Buffer Full -

523126 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www



unit-3-1

:*********************************************************
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553566 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523522 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
```

```
capture CAPO_RH type raw-data

reinject-hide

 interface

OUTSIDE

 [Buffer Full -

523432 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
```

Observation 2. Analyse de l'indicateur de connexion pour le flux avec le port source 59210.

<#root>

```
firepower#

cluster exec show conn addr 192.168.240.51



unit-1-1

(LOCAL):*******************************************************
25 in use, 102 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 2 in use, 122 most used
centralized connections: 0 in use, 39 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:

59210

, idle 0:00:03, bytes 0,

flags Y



unit-2-1

:*************************************************************
21 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 0 in use, 28 most used
centralized connections: 0 in use, 14 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
```

```
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:

59210

, idle 0:00:00, bytes 610132872,

flags b N
```

```
unit-3-1

:************************************************************
19 in use, 55 most used
Cluster:
fwd connections: 1 in use, 5 most used
dir connections: 0 in use, 127 most used
centralized connections: 0 in use, 24 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.240.51:80 NP Identity Ifc 192.168.240.50:

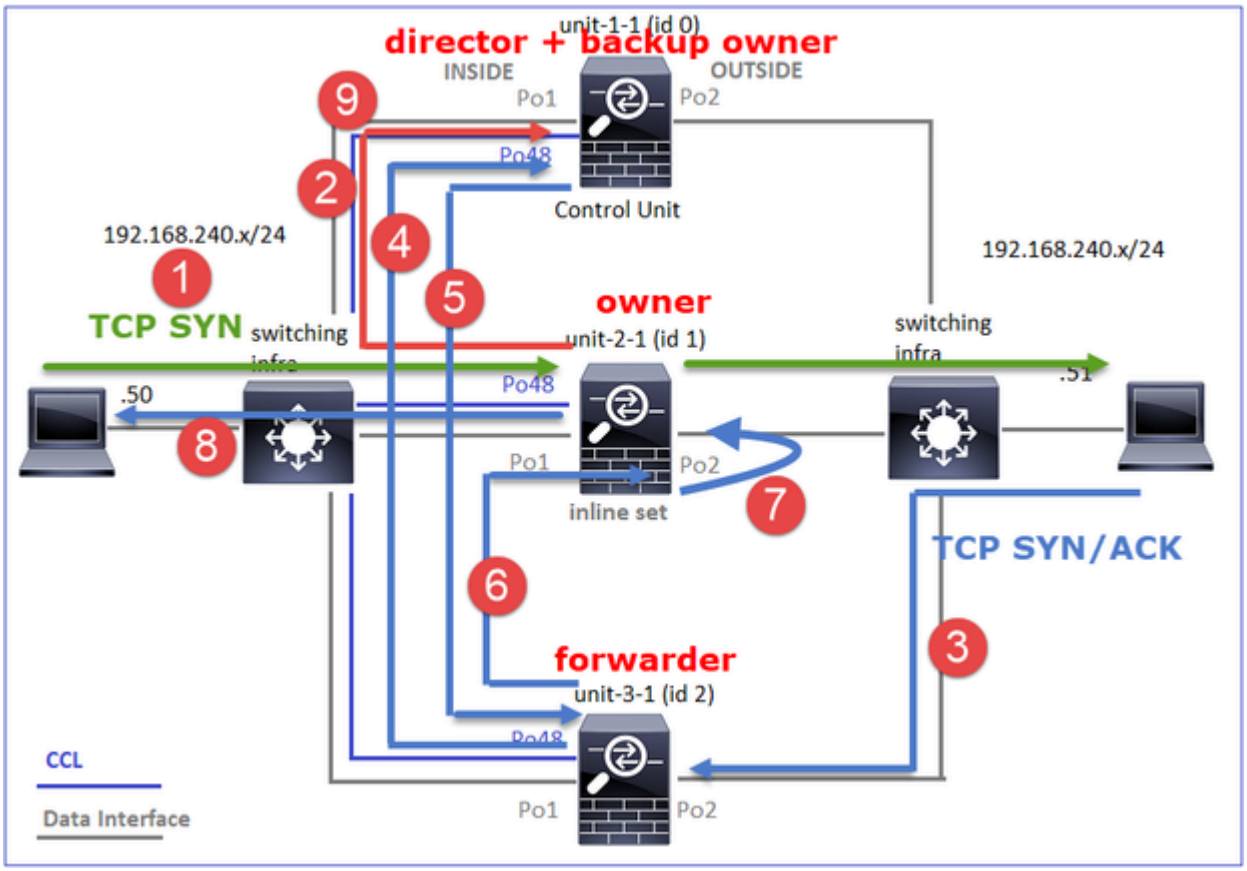
59210

, idle 0:00:00, bytes 0,

flags z
```

Unité	Drapeau	Remarque
Unité-1-1	O	· Directeur/propriétaire de sauvegarde
Unité-2-1	b N	· Flow Owner - L'unité gère le flux
Unité-3-1	z	· Transporteur

Cela peut être visualisé comme suit :



1. Le paquet SYN TCP arrive de l'hôte A à l'unité 2-1. L'unité 2-1 devient le propriétaire du flux et l'unité 1-1 est élue comme directeur
2. Unit-1-1 est élu propriétaire de sauvegarde (puisqu'il s'agit du directeur). Le propriétaire du flux envoie un message de monodiffusion « cluster add » sur UDP 4193 à. informer le propriétaire de la sauvegarde du flux.
3. Le paquet TCP SYN/ACK arrive de l'hôte B à l'unité 3-1. Le flux est asymétrique.
4. L'unité 3-1 transmet le paquet via la CCL au directeur (unité 1-1).
5. L'unité 1-1 (directeur) sait que le propriétaire est l'unité 2-1, renvoie le paquet au redirecteur (unité 3-1) et l'informe que le propriétaire est l'unité 2-1.
6. Unit-3-1 envoie le paquet à Unit-2-1 (propriétaire).
7. Unit-2-1 réinjecte le paquet sur l'interface OUTSIDE.
8. L'unité 2-1 transfère le paquet vers l'hôte A.
9. Une fois la connexion terminée, le propriétaire envoie un message de suppression de cluster pour supprimer les informations de flux du propriétaire de sauvegarde.

Remarque : Il est important que l'étape 2 (paquet transitant par la CCL) ait lieu avant l'étape 4 (trafic de données). Dans un autre cas (par exemple, la condition de race), le directeur n'est pas au courant du flux. Ainsi, puisqu'il s'agit d'un ensemble en ligne, transfère le paquet vers sa destination. Si les interfaces ne sont pas dans un ensemble en ligne, le paquet de données est abandonné.

Observation 3. La capture avec suivi montre le trafic asymétrique et les échanges sur la CCL :

Transfert du trafic (TCP SYN)

Unit-2-1 (propriétaire) :

<#root>

```
firepower#
cluster exec unit unit-2-1 show cap CAPI packet-number 1 trace


1: 09:19:49.760702 192.168.240.50.59210 > 192.168.240.51.80: S 4110299695:4110299695(0) win 29200 <mss
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (1) got initial, attempting ownership.


Phase: 2
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (1) am becoming owner
```

Trafic de retour (TCP SYN/ACK)

Unit-3-1 (ID 2 - redirecteur) envoie le paquet via la CCL à unit-1-1 (ID 0 - directeur) :

<#root>

```
firepower#
cluster exec unit unit-3-1 show cap CAPO packet-number 1 trace


1: 09:19:49.760336 192.168.240.51.80 > 192.168.240.50.59210:
S
 4209225081:4209225081(0)
ack
 4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,wscale 7>
```

```
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW
```

I (2) am asking director (0).

Unit-1-1 (directeur) - Unit-1-1 (ID 0) sait que le propriétaire du flux est unit-2-1 (ID 1) et renvoie le paquet sur la CCL à unit-3-1 (ID 2 - redirecteur) :

<#root>

```
firepower#
```

cluster exec show cap CAPO packet-number 1 trace

```
unit-1-1(LOCAL):*******************************************************
```

```
1: 09:19:49.761038 192.168.240.51.80 > 192.168.240.50.59210:
```

S

```
 4209225081:4209225081(0)
```

ack

```
 4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,wscale 7>
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: STUB
```

I (0) am director, valid owner (1), update sender (2).

Unit-3-1 (ID 2 - redirecteur) obtient le paquet via la CCL et l'envoie à Unit-2-1 (ID 1 - propriétaire) :

<#root>

```
firepower#
```

cluster exec unit unit-3-1 show cap CAPO packet-number 2 trace

```
...
2: 09:19:49.761008 192.168.240.51.80 > 192.168.240.50.59210:
```

```
S

 4209225081:4209225081(0) ack 4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,w
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: STUB


I (2) am becoming forwarder to (1), sender (0).
```

Le propriétaire réinjecte et transfère le paquet vers la destination :

<#root>

```
firepower#

cluster exec unit unit-2-1 show cap CAPO packet-number 2 trace



2: 09:19:49.775701 192.168.240.51.80 > 192.168.240.50.59210:

S

 4209225081:4209225081(0)

ack

 4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,wscale 7>
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: FULL


I (1) am owner, sender (2).
```

Observation 4. Les syslog du plan de données FTD indiquent la création et la fin de la connexion sur toutes les unités :

- Unité-1-1 (directeur/propriétaire de la sauvegarde)
- Unit-2-1 (propriétaire)
- Unité-3-1 (redirecteur)

<#root>

```
firepower#
```

```
 cluster exec show log | i 59210


unit-1-1(LOCAL):*******************************************************
Dec 03 2020 09:19:49: %FTD-6-302022:

Built director stub TCP connection

 for INSIDE:192.168.240.50/59210 (192.168.240.50/59210) to OUTSIDE:192.168.240.51/80 (192.168.240.51/80)
Dec 03 2020 09:19:59: %FTD-6-302023:

Teardown director TCP connection

 for INSIDE:192.168.240.50/59210 to OUTSIDE:192.168.240.51/80 duration 0:00:09 forwarded bytes 0 Cluste

unit-2-1:*******************************************************
Dec 03 2020 09:19:49: %FTD-6-302303:

Built TCP state-bypass connection

 14483 from INSIDE:192.168.240.50/59210 (192.168.240.50/59210) to OUTSIDE:192.168.240.51/80 (192.168.24
Dec 03 2020 09:19:59: %FTD-6-302304:

Teardown TCP state-bypass connection

 14483 from INSIDE:192.168.240.50/59210 to OUTSIDE:192.168.240.51/80 duration 0:00:09 bytes 1024003336

unit-3-1:*******************************************************
Dec 03 2020 09:19:49: %FTD-6-302022:

Built forwarder stub TCP connection

 for OUTSIDE:192.168.240.51/80 (192.168.240.51/80) to unknown:192.168.240.50/59210 (192.168.240.50/5921
Dec 03 2020 09:19:59: %FTD-6-302023:

Teardown forwarder TCP connection

 for OUTSIDE:192.168.240.51/80 to unknown:192.168.240.50/59210 duration 0:00:09 forwarded bytes 1024003
```

Dépannage

Présentation du dépannage de cluster

Les problèmes de cluster peuvent être classés en :

- Problèmes liés au plan de contrôle (problèmes liés à la stabilité du cluster)
- Problèmes liés au plan de données (problèmes liés au trafic de transit)

Problèmes de plan de données de cluster

Problèmes courants de NAT/PAT

Considérations de configuration importantes

- Les pools de traduction d'adresses de port (PAT) doivent avoir au moins autant d'adresses IP disponibles que le nombre d'unités dans le cluster, de préférence plus d'adresses IP que de noeuds de cluster.

- Les commandes xlate per-session par défaut doivent être conservées sauf s'il existe une raison spécifique de les désactiver. Tout xlate PAT construit pour une connexion dont xlate par session est désactivé est toujours géré par l'unité de noeud de contrôle dans le cluster, ce qui peut entraîner une dégradation des performances.

Utilisation élevée de la plage du pool PAT en raison du trafic provenant de ports faibles qui entraîne un déséquilibre IP de cluster

Le FTD divise une adresse IP PAT en plages et tente de maintenir xlate dans la même plage source. Ce tableau montre comment un port source est converti en un port global dans la même plage source.

Port Src D'Origine	Port Src Traduit
1-511	1-511
512-1023	512-1023
1024-65535	1024-65535

Lorsqu'une plage de ports source est pleine et qu'un nouvel xlate PAT doit être alloué à partir de cette plage, FTD passe à l'IP suivante pour allouer de nouvelles traductions pour cette plage de ports source.

Symptômes :

Problèmes de connectivité pour le trafic NAT qui traverse le cluster

Vérification :

<#root>

#

show nat pool

Les journaux du plan de données FTD indiquent l'épuisement du pool PAT :

<#root>

Dec 9 09:00:00 192.0.2.10 FTD-FW %ASA-3-202010:

PAT pool exhausted. Unable to create TCP connection

from Inside:192.0.2.150/49464 to Outside:192.0.2.250/20015

```
Dec 9 09:00:00 192.0.2.10 FTD-FW %ASA-3-202010:
PAT pool exhausted. Unable to create TCP connection
 from Inside:192.0.2.148/54141 to Outside:192.0.2.251/443
```

Atténuation :

Configurez la plage de ports plats NAT et incluez les ports de réserve.

En outre, dans les versions postérieures à la version 6.7/9.15.1, vous pouvez vous retrouver avec une distribution de bloc de port déséquilibrée uniquement lorsque les noeuds quittent/rejoignent le cluster avec un trafic d'arrière-plan énorme qui est soumis à la PAT. La seule façon de le récupérer par lui-même est lorsque les blocs de ports sont libérés pour être redistribués sur les noeuds.

Avec la distribution basée sur les blocs de ports, lorsqu'un noeud est alloué avec, par exemple, 10 blocs de ports tels que pb-1, pb-2 ... pb-10. Le noeud commence toujours par le premier bloc de ports disponible et lui alloue un port aléatoire jusqu'à ce qu'il s'épuise. L'allocation passe au bloc de ports suivant uniquement lorsque tous les blocs de ports jusqu'à ce point sont épuisés.

Par exemple, si un hôte établit 512 connexions, l'unité alloue des ports mappés pour toutes ces 512 connexions de pb-1 de manière aléatoire. Maintenant, avec toutes ces 512 connexions actives, quand l'hôte établit la 513e connexion depuis que pb-1 est épuisé, il passe à pb-2 et lui alloue un port aléatoire. Là encore, sur 513 connexions, supposons que la 10e connexion est terminée et a effacé un port disponible dans pb-1. À ce stade, si l'hôte établit la 514e connexion, l'unité de cluster alloue un port mappé à partir de pb-1 et non de pb-2, car pb-1 dispose désormais d'un port libre (qui a été libéré dans le cadre de la suppression de la 10e connexion).

L'élément important à garder à l'esprit est que l'allocation se produit à partir du premier bloc de ports disponible avec des ports libres de sorte que les derniers blocs de ports sont toujours disponibles pour la redistribution dans un système normalement chargé. En outre, la PAT est généralement utilisée pour les connexions de courte durée. La probabilité qu'un bloc de ports devienne disponible dans un délai plus court est très élevée. Ainsi, le temps nécessaire pour que la distribution de pool soit équilibrée peut être amélioré avec la distribution de pool basée sur des blocs de ports.

Cependant, si tous les blocs de ports, de pb-1 à pb-10, sont épuisés ou si chaque bloc de ports contient un port pour une connexion de longue durée, les blocs de ports ne sont jamais libérés rapidement et redistribués. Dans ce cas, l'approche la moins perturbatrice consiste à :

1. Identifiez les noeuds avec des blocs de ports excessifs (show nat pool cluster summary).
2. Identifiez les blocs de ports les moins utilisés sur ce noeud (show nat pool ip <addr> detail).
3. Effacer les xlate pour ces blocs de ports (clear xlate global <addr> gport 'start-end') pour les rendre disponibles pour la redistribution.

⚠ Avertissement : Cela interrompt les connexions concernées.

Impossible de naviguer vers des sites Web à deux canaux (comme la messagerie Web, les services bancaires, etc.) ou vers des sites Web SSO lorsque la redirection vers une autre destination se produit.

Symptômes :

Impossible de naviguer vers des sites Web à deux canaux (tels que webmail, sites Web bancaires, etc.). Lorsqu'un utilisateur se connecte à un site Web qui exige que le client ouvre une seconde connexion/connexion et que la seconde connexion est hachée vers un membre du cluster différent de celui qui a reçu la première connexion hachée, et que le trafic utilise un pool PAT IP, le trafic est réinitialisé par le serveur lorsqu'il reçoit la connexion d'une adresse IP publique différente.

Vérification :

Effectuez des captures de cluster de plan de données pour voir comment le flux de transit affecté est géré. Dans ce cas, une réinitialisation TCP provient du site Web de destination.

Atténuation (avant 6.7/9.15.1) :

- Vérifiez si des applications multisessions utilisent plusieurs adresses IP mappées.
- Utilisez la commande show nat pool cluster summary pour vérifier si le pool est distribué de manière égale.
- Utilisez la commande cluster exec show conn pour vérifier si le trafic est équilibré correctement en charge.
- Utilisez la commande show nat pool cluster ip <address>detail pour vérifier l'utilisation du pool d'adresses IP rémanentes.
- Activez syslog 305021 (6.7/9.15) pour voir quelles connexions n'ont pas pu utiliser l'adresse IP rémanente.
- Pour résoudre le problème, ajoutez d'autres adresses IP au pool PAT ou affinez l'algorithme d'équilibrage de charge sur les commutateurs connectés.

À propos de l'algorithme d'équilibrage de charge EtherChannel :

- Pour les modèles autres que FP9300 et si l'authentification s'effectue via un serveur : Réglez l'algorithme d'équilibrage de charge d'éther-channel sur le commutateur adjacent de Source IP/Port et Destination IP/Port à Source IP et Destination IP.
- Pour les modèles non FP9300 et si l'authentification s'effectue via plusieurs serveurs : Réglez l'algorithme d'équilibrage de charge d'éther-channel sur le commutateur adjacent de Source IP/Port et Destination IP/Port à Source IP.
- Pour FP9300 : Sur le châssis FP9300, l'algorithme d'équilibrage de charge est défini comme source-dest-port source-dest-ip source-dest-mac et ne peut pas être modifié. La solution de contournement, dans ce cas, est d'utiliser FlexConfig pour ajouter des commandes xlate per-session deny à la configuration FTD pour forcer le trafic pour certaines adresses IP de destination (pour les applications problématiques/incompatibles) à être géré uniquement par le noeud de contrôle dans le cluster intra-châssis. La solution de contournement s'accompagne des effets secondaires suivants :
    - Aucun équilibrage de charge du trafic traduit différemment (tout va au noeud de

contrôle).
- Possibilité que les slots xlate soient épuisés (et affecter négativement la traduction NAT pour le reste du trafic sur le noeud de contrôle).
- Évolutivité réduite du cluster intra-châssis.

Faibles performances du cluster en raison de l'ensemble du trafic envoyé au noeud de contrôle en raison d'un nombre insuffisant d'adresses IP PAT dans les pools.

Symptômes :

Il n'y a pas assez d'adresses IP PAT dans le cluster pour allouer une adresse IP libre aux noeuds de données, et par conséquent, tout le trafic soumis à la configuration PAT est transféré au noeud de contrôle pour traitement.

Vérification :

Utilisez la commande show nat pool cluster pour afficher les allocations pour chaque unité et confirmer qu'elles possèdent toutes au moins une adresse IP dans le pool.

Atténuation :

Pour les versions antérieures à 6.7/9.15.1, assurez-vous que vous disposez d'un pool PAT d'une taille au moins égale au nombre de noeuds dans le cluster. Dans les versions postérieures à 6.7/9.15.1 avec pool PAT, vous allouez des blocs de ports à partir de toutes les adresses IP du pool PAT. Si l'utilisation du pool PAT est très élevée, ce qui entraîne un épuisement fréquent du pool, vous devez augmenter la taille du pool PAT (voir la section FAQ).

Performances faibles en raison de l'ensemble du trafic envoyé au noeud de contrôle, car les xlate ne sont pas activés par session.

Symptômes :

De nombreux flux de sauvegarde UDP à haut débit sont traités via le noeud de contrôle de cluster, ce qui peut avoir un impact sur les performances.

Contexte :

Seules les connexions qui utilisent des xlate activés par session peuvent être traitées par un noeud de données qui utilise la PAT. Utilisez la commande show run all xlate pour afficher la configuration xlate par session.

L'activation par session signifie que l'algorithme xlate est désactivé immédiatement lorsque la connexion associée est interrompue. Cela permet d'améliorer les performances de connexion par seconde lorsque les connexions sont soumises à la PAT. Les extractions non par session sont actives pendant 30 secondes supplémentaires après l'interruption de la connexion associée, et si le débit de connexion est suffisamment élevé, les 65 000 ports TCP/UDP disponibles sur chaque adresse IP globale peuvent être utilisés en peu de temps.

Par défaut, tout le trafic TCP est activé par xlate et seul le trafic DNS UDP est activé par session.

Cela signifie que tout le trafic UDP non-DNS est transféré au noeud de contrôle pour traitement.

Vérification :

Utilisez cette commande pour vérifier la connexion et la distribution des paquets entre les unités de cluster :

<#root>

firepower#

show cluster info conn-distribution

firepower#

show cluster info packet-distribution

firepower#

show cluster info load-monitor

Utilisez la commande cluster exec show conn pour voir quels noeuds de cluster possèdent les connexions UDP.

<#root>

firepower#

cluster exec show conn

Utilisez cette commande pour comprendre l'utilisation du pool sur les noeuds du cluster.

<#root>

firepower#

cluster exec show nat pool ip

| in UDP

Atténuation :

Configurez la PAT par session (commande per-session permit udp) pour le trafic d'intérêt (par exemple, UDP). Pour ICMP, vous ne pouvez pas modifier la PAT multisession par défaut. Par conséquent, le trafic ICMP est toujours géré par le noeud de contrôle lorsque la PAT est configurée.

La distribution du pool PAT devient déséquilibrée lorsque les noeuds quittent/rejoignent le cluster.

Symptômes :

- Les problèmes de connectivité dus à l'allocation IP PAT peuvent devenir déséquilibrés dans le temps en raison des unités qui quittent le cluster et y rejoignent le cluster.
- Dans les versions post-6.7/9.15.1, il peut y avoir des cas où le noeud nouvellement joint ne peut pas obtenir suffisamment de blocs de ports. Un noeud qui n'a pas de bloc de port redirige le trafic vers le noeud de contrôle. Un noeud qui possède au moins un bloc de ports gère le trafic et le supprime une fois le pool épuisé.

Vérification :

- Les syslogs du plan de données affichent des messages tels que :

<#root>

```
%ASA-3-202010:
NAT pool exhausted. Unable to create TCP connection
 from inside:192.0.2.1/2239 to outside:192.0.2.150/80
```

- Utilisez la commande show nat pool cluster summary pour identifier la distribution du pool.
- Utilisez la commande cluster exec show nat pool ip <addr>detail pour comprendre l'utilisation du pool sur les noeuds du cluster.

Atténuation :

- Pour les versions antérieures à 6.7/9.15.1, quelques solutions de contournement sont décrites dans l'ID de bogue Cisco CSCvd10530
- Dans les versions postérieures à la version 6.7/9.15.1, utilisez la commande clear xlate global <ip> gport <start-end> pour effacer manuellement certains des blocs de ports sur d'autres noeuds en vue d'une redistribution vers les noeuds requis.

Symptômes :

Principaux problèmes de connectivité pour le trafic qui est PATé par le cluster. En effet, le plan de données FTD, par conception, n'envoie pas de GARP pour les adresses NAT globales.

Vérification :

La table ARP des périphériques connectés directement affiche différentes adresses MAC de l'interface de données de cluster après un changement du noeud de contrôle :

<#root>

```
root@kali2:~/tests#
```

arp -a

```
? (192.168.240.1) at f4:db:e6:
```

33:44:2e

```
 [ether] on eth0
root@kali2:~/tests#
```

arp -a

```
? (192.168.240.1) at f4:db:e6:
```

9e:3d:0e

```
 [ether] on eth0
```

Atténuation :

Configurez l'adresse MAC statique (virtuelle) sur les interfaces de données de cluster.

Les connexions soumises à la PAT échouent

Symptômes :

Problèmes de connectivité pour le trafic qui est PATé par le cluster.

Vérification/atténuation :

- Assurez-vous que la configuration est correctement répliquée.
- Assurez-vous que le pool est réparti uniformément.
- Assurez-vous que la propriété du pool est valide.
- Aucun incrément de compteur d'échec dans show asp cluster counter.
- Assurez-vous que les flux de directeur/redirecteur sont créés avec les informations appropriées.
- Validez si les xlate de sauvegarde sont créés, mis à jour et nettoyés comme prévu.
- Validez si les xlate sont créés et terminés selon le comportement « par session ».
- Activez « debug nat 2 » pour une indication des erreurs éventuelles. Notez que cette sortie peut être très bruyante, par exemple :

<#root>

```
firepower#
```

debug nat 2

```
nat:
```

no free blocks available to reserve for 192.168.241.59, proto 17

```
nat: no free blocks available to reserve for 192.168.241.59, proto 17
nat: no free blocks available to reserve for 192.168.241.58, proto 17
nat: no free blocks available to reserve for 192.168.241.58, proto 17
nat: no free blocks available to reserve for 192.168.241.57, proto 17
```

Pour arrêter le débogage :

<#root>

```
firepower#
```

un all

- Activez la connexion et les syslogs liés à la NAT pour mettre en corrélation les informations avec une connexion défaillante.

Améliorations PAT de mise en grappe ASA et FTD (post-9.15 et 6.7)

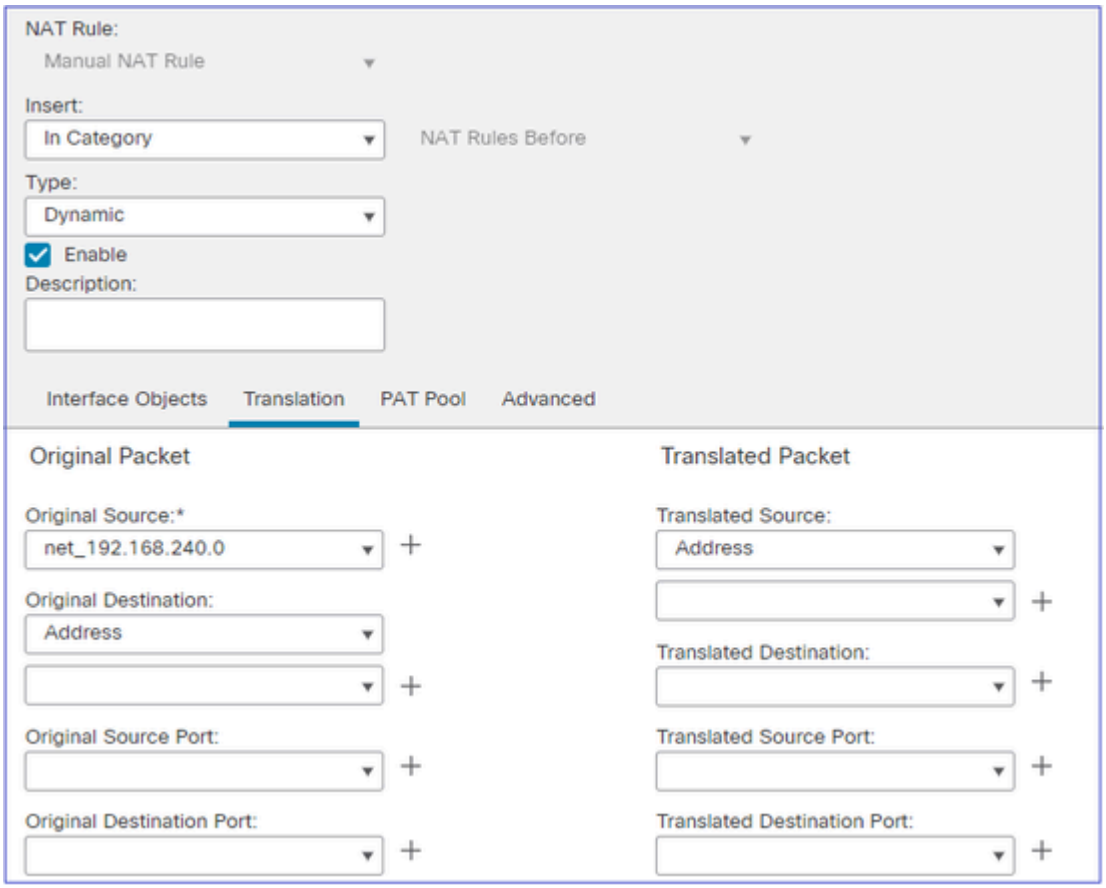
Qu'est-ce qui a changé ? L'opération PAT a été repensée. Les adresses IP individuelles ne sont plus distribuées à chacun des membres du cluster. Au lieu de cela, les adresses IP PAT sont divisées en blocs de ports et distribuées ces blocs de ports de manière égale (autant que possible) entre les membres du cluster, en combinaison avec le fonctionnement de collage IP.

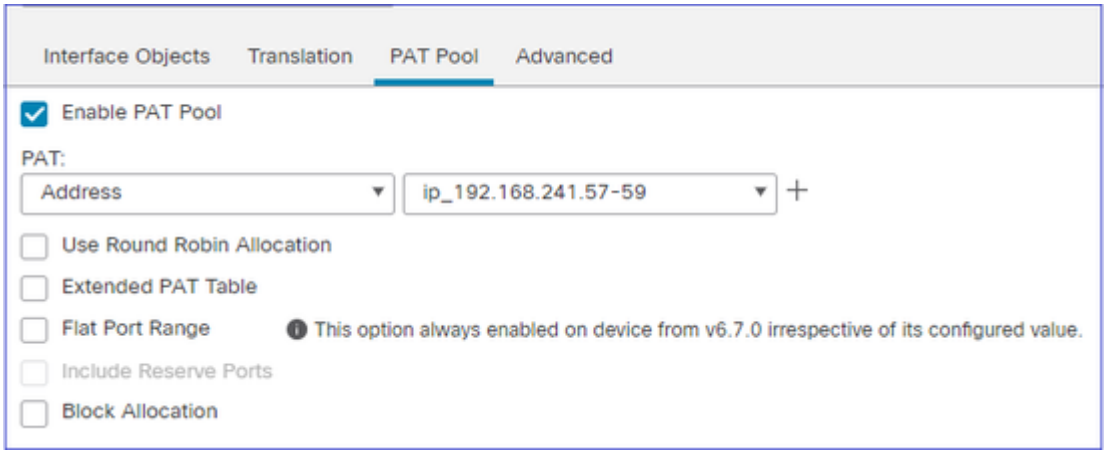
La nouvelle conception répond à ces limitations (voir la section précédente) :

- Les applications multisessions sont affectées en raison d'un manque d'adhérence IP au niveau de la grappe.
- La condition est d'avoir un pool PAT de taille au moins égale au nombre de noeuds dans le cluster.
- La distribution du pool PAT devient déséquilibrée lorsque les noeuds quittent/rejoignent le cluster.
- Aucun syslog pour indiquer un déséquilibre du pool PAT.

Techniquement, au lieu des plages de ports par défaut 1-511, 512-1023 et 1024-65535, il y a maintenant 1024-65535 comme plage de ports par défaut pour la PAT. Cette plage par défaut peut être étendue pour inclure la plage de ports privilégiés 1-1023 pour la PAT standard (option include-reserve).

Voici un exemple de configuration de pool PAT sur FTD 6.7. Pour plus d'informations, consultez la section correspondante du Guide de configuration :





Informations de dépannage supplémentaires sur la PAT

Syslogs du plan de données FTD (post-6.7/9.15.1)

Un syslog d'invalidation de la rémanence est généré lorsque tous les ports sont épuisés dans l'IP rémanente sur un noeud de cluster, et l'allocation passe à l'IP disponible suivante avec des ports libres, par exemple :

```
%ASA-4-305021: Ports exhausted in pre-allocated PAT pool IP 192.0.2.100 for host 198.51.100.100 Allocat
```

Un syslog de déséquilibre de pool est généré sur un noeud lorsqu'il rejoint le cluster et n'obtient aucune part ou une part inégale de blocs de ports, par exemple :

```
%ASA-4-305022: Cluster unit ASA-4 has been allocated 0 port blocks for PAT usage. All units should have
%ASA-4-305022: Cluster unit ASA-4 has been allocated 12 port blocks for PAT usage. All units should hav
```

Commandes show

État de distribution du pool :

Dans le résultat show nat pool cluster summary, pour chaque adresse IP PAT, il ne doit pas y avoir de différence de plus d'un bloc de ports entre les noeuds dans un scénario de distribution équilibrée. Exemples de distribution équilibrée et déséquilibrée de blocs de ports.

<#root>

```
firepower#
```

show nat pool cluster summary

```
port-blocks count display order: total, unit-1-1, unit-2-1, unit-3-1
IP OUTSIDE:ip_192.168.241.57-59 192.168.241.57 (126 -
```

42 / 42 / 42

```
)
IP OUTSIDE:ip_192.168.241.57-59 192.168.241.58 (126 - 42 / 42 / 42)
IP OUTSIDE:ip_192.168.241.57-59 192.168.241.59 (126 - 42 / 42 / 42)
```

Distribution déséquilibrée :

<#root>

```
firepower#
```

show nat pool cluster summary

```
port-blocks count display order: total, unit-1-1, unit-4-1, unit-2-1, unit-3-1
IP outside:src_map 192.0.2.100 (128 - 32 /
```

22 / 38

```
 / 36)
```

Statut de propriété du pool :

Dans le résultat de la commande show nat pool cluster, il ne doit pas y avoir un seul bloc de ports avec le propriétaire ou la sauvegarde comme UNKNOWN. S'il y en a un, cela indique un problème de communication de propriété de pool. Exemple :

```
<#root>

firepower#

show nat pool cluster | in


[3072-3583], owner unit-4-1, backup <

UNKNOWN

>
[56832-57343], owner <UNKNOWN>, backup <UNKNOWN>
[10240-10751], owner unit-2-1, backup <UNKNOWN>
```

Comptabilisation des allocations de ports dans les blocs de ports :

La commande show nat pool est améliorée avec des options supplémentaires pour afficher des informations détaillées ainsi que la sortie filtrée.

Exemple :

```
<#root>

firepower#

show nat pool detail


TCP PAT pool INSIDE, address 192.168.240.1, range 1-1023, allocated 0
TCP PAT pool INSIDE, address 192.168.240.1, range 1024-65535, allocated 18
UDP PAT pool INSIDE, address 192.168.240.1, range 1-1023, allocated 0
UDP PAT pool INSIDE, address 192.168.240.1, range 1024-65535, allocated 20
TCP PAT pool OUTSIDE, address 192.168.241.1, range 1-1023, allocated 0
TCP PAT pool OUTSIDE, address 192.168.241.1, range 1024-65535, allocated 18
UDP PAT pool OUTSIDE, address 192.168.241.1, range 1-1023, allocated 0
UDP PAT pool OUTSIDE, address 192.168.241.1, range 1024-65535, allocated 20
UDP PAT pool OUTSIDE, address 192.168.241.58
range 1024-1535, allocated 512
range 1536-2047, allocated 512
range 2048-2559, allocated 512
range 2560-3071, allocated 512
...
unit-2-1:**************************************************************
UDP PAT pool OUTSIDE, address 192.168.241.57
range 1024-1535, allocated 512 *
range 1536-2047, allocated 512 *
range 2048-2559, allocated 512 *
```

« * » indique qu'il s'agit d'un bloc de ports sauvegardé

Pour résoudre ce problème, utilisez la commande clear xlate global <ip> gport <start-end> pour effacer manuellement certains des blocs de ports sur d’autres noeuds pour les redistribuer aux noeuds requis.

Redistribution déclenchée manuellement des blocs de ports :

- Dans un réseau de production avec un trafic constant, lorsqu'un noeud quitte et rejoint le cluster (probablement en raison d'un retour arrière), il peut arriver qu'il ne puisse pas obtenir une part égale du pool ou, dans le pire des cas, qu'il ne puisse obtenir aucun bloc de port.
- Utilisez la commande show nat pool cluster summary pour identifier quel noeud possède plus de blocs de ports que nécessaire.
- Sur les noeuds qui possèdent plus de blocs de ports, utilisez la commande show nat pool ip <addr> detail pour déterminer les blocs de ports avec le plus petit nombre d'allocations.
- Utilisez la commande clear xlate global <address> gport <start-end> pour effacer les traductions créées à partir de ces blocs de ports afin qu'elles deviennent disponibles pour la redistribution vers les noeuds requis, par exemple :

<#root>

```
firepower#
```

show nat pool detail | i 19968

```
        range 19968-20479, allocated 512
        range 19968-20479, allocated 512
        range 19968-20479, allocated 512

firepower#
```

clear xlate global 192.168.241.57 gport 19968-20479

```
INFO: 1074 xlates deleted
```

Foire aux questions (FAQ) pour la PAT post-6.7/9.15.1

Q. Si vous disposez du nombre d'IP disponibles pour le nombre d'unités disponibles dans le cluster, pouvez-vous toujours utiliser 1 IP par unité en option ?

R. Plus maintenant, et il n'y a pas de basculement entre les schémas de distribution de pools basés sur des adresses IP et les schémas de distribution de pools basés sur des blocs de ports.

L'ancien système de distribution de pool basé sur des adresses IP a entraîné des pannes d'applications multisessions où plusieurs connexions (qui font partie d'une seule transaction d'application) d'un hôte sont équilibrées en charge sur différents noeuds du cluster et ainsi traduites par différentes adresses IP mappées qui conduisent au serveur de destination à les voir comme provenant de différentes entités.

De plus, avec le nouveau schéma de distribution basé sur les blocs de ports, même si vous pouvez désormais utiliser une adresse IP PAT unique, il est toujours recommandé d'avoir suffisamment d'adresses IP PAT en fonction du nombre de connexions requises pour la PAT.

Q. Pouvez-vous toujours disposer d'un pool d'adresses IP pour le pool PAT du cluster ?

R. Oui, vous pouvez. Les blocs de ports de toutes les adresses IP du pool PAT sont distribués sur les noeuds du cluster.

Q. Si vous utilisez un certain nombre d'adresses IP pour le pool PAT, le même bloc de ports est-il attribué à chaque membre par adresse IP ?

R. Non, chaque adresse IP est distribuée indépendamment.

Q. Tous les noeuds de cluster ont toutes les adresses IP publiques, mais seulement un sous-ensemble de ports ? Si tel est le cas, est-il alors garanti que chaque fois que l'adresse IP source utilise la même adresse IP publique ?

R. C'est exact, chaque adresse IP PAT est partiellement détenue par chaque noeud. Si une adresse IP publique choisie est épuisée sur un noeud, un syslog est généré qui indique que l'adresse IP rémanente ne peut pas être conservée et l'allocation passe à l'adresse IP publique disponible suivante. Qu'il s'agisse d'un déploiement autonome, à haute disponibilité ou en cluster, la fidélité IP est toujours assurée au mieux en fonction de la disponibilité du pool.

Q. Tout repose-t-il sur une adresse IP unique dans le pool PAT, mais ne s'applique pas si plusieurs adresses IP du pool PAT sont utilisées ?

R. Elle s'applique également à plusieurs adresses IP dans le pool PAT. Les blocs de ports de chaque IP du pool PAT sont distribués sur les noeuds de cluster. Chaque adresse IP du pool PAT est partagée entre tous les membres du cluster. Ainsi, si vous avez une classe C d'adresses dans le pool PAT, chaque membre du cluster possède des pools de ports de chacune des adresses du pool PAT.

Q. Fonctionne-t-il avec CGNAT ?

R. Oui, la TCNAG est aussi bien prise en charge. CGNAT, également connu sous le nom de PAT d'allocation de blocs, a une taille de bloc par défaut de '512' qui peut être modifiée par l'intermédiaire de l'interface CLI de taille d'allocation de blocs xlate. Dans le cas d'une PAT dynamique régulière (non CGNAT), la taille de bloc est toujours '512', ce qui est fixe et non configurable.

Q. Si l'unité quitte le cluster, le noeud de contrôle attribue-t-il la plage de blocs de ports aux autres unités ou la conserve-t-il pour elle-même ?

R. Chaque bloc de ports a un propriétaire et une sauvegarde. Chaque fois qu'un xlate est créé à partir d'un bloc de ports, il est également répliqué sur le noeud de sauvegarde du bloc de ports. Lorsqu'un noeud quitte le cluster, le noeud de secours possède tous les blocs de ports et toutes les connexions en cours. Le noeud de sauvegarde, depuis qu'il est devenu le propriétaire de ces blocs de ports supplémentaires, sélectionne une nouvelle sauvegarde pour eux et réplique tous

les xlate actuels sur ce noeud pour gérer les scénarios d'échec.

Q. Sur la base de cette alerte, quelles mesures peuvent être prises pour renforcer l'adhérence ?

R. Il y a deux raisons possibles pour lesquelles l'adhésivité ne peut pas être préservée.

Raison-1 : La charge du trafic est incorrectement équilibrée, ce qui explique que l'un des noeuds voit un nombre de connexions plus élevé que les autres, ce qui entraîne l'épuisement de l'adresse IP rémanente. Vous pouvez résoudre ce problème si vous vous assurez que le trafic est réparti de manière égale entre les noeuds de cluster. Par exemple, sur un cluster FPR41xx, modifiez l'algorithme d'équilibrage de charge sur les commutateurs connectés. Sur un cluster FPR9300, assurez-vous que le nombre de lames sur le châssis est identique.

Raison 2 : L'utilisation du pool PAT est très élevée, ce qui entraîne un épuisement fréquent du pool. Pour résoudre ce problème, augmentez la taille du pool PAT.

Q. Comment la prise en charge du mot clé extended est-elle gérée ? Affiche-t-il une erreur et empêche-t-il l'ajout de toute la commande NAT pendant la mise à niveau, ou supprime-t-il le mot clé étendu et affiche-t-il un avertissement ?

R. L'option étendue PAT n'est pas prise en charge dans Cluster à partir de ASA 9.15.1/FP 6.7. L'option de configuration n'est supprimée d'aucune interface CLI/ASDM/CSM/FMC. Une fois configurée (directement ou indirectement par le biais d'une mise à niveau), un message d'avertissement s'affiche et la configuration est acceptée, mais vous ne voyez pas la fonctionnalité étendue de la PAT en action.

Q. Est-ce le même nombre de traductions que des connexions simultanées ?

R. Dans les versions antérieures à la version 6.7/9.15.1, bien qu'il soit compris entre 1 et 65535, comme les ports source ne sont jamais très utilisés dans la plage 1-1024, il est en fait compris entre 1024 et 65535 (64512 cons). Dans l'implémentation post-6.7/9.15.1 avec 'flat' comme comportement par défaut, c'est 1024-65535. Mais si vous voulez utiliser le 1-1024, vous pouvez avec l'option "include-reserve".

Q. Si le noeud rejoint le cluster, l’ancien noeud de sauvegarde est utilisé comme sauvegarde et ce noeud de sauvegarde lui donne son ancien bloc de ports ?

R. Cela dépend de la disponibilité des blocs de ports à ce moment-là. Lorsqu'un noeud quitte le cluster, tous ses blocs de ports sont déplacés vers le noeud de sauvegarde. C'est alors le noeud de contrôle qui accumule les blocs de ports libres et les distribue aux noeuds requis.

Q. Si l’état du noeud de contrôle change, si un nouveau noeud de contrôle est sélectionné, si l’allocation de blocs PAT doit être maintenue ou si les blocs de ports sont réalloués en fonction du nouveau noeud de contrôle ?

R. Le nouveau noeud de contrôle comprend quels blocs ont été alloués et lesquels sont libres et commence à partir de là.

Q. Le nombre maximal d'xlate est-il le même que le nombre maximal de connexions simultanées

avec ce nouveau comportement ?

R. Oui. Le nombre maximal d'xlates dépend de la disponibilité des ports PAT. Cela n'a rien à voir avec le nombre maximal de connexions simultanées. Si vous n'autorisez qu'une seule adresse, vous avez 65535 connexions possibles. Si vous en avez besoin, vous devez allouer plus d'adresses IP. S'il y a suffisamment d'adresses/de ports, vous pouvez atteindre le nombre maximal de connexions simultanées.

Q. Quel est le processus d'allocation de bloc de ports lorsqu'un nouveau membre de cluster est ajouté ? Que se passe-t-il si un membre du cluster est ajouté en raison d'un redémarrage ?

R. Les blocs de ports sont toujours distribués par le noeud de contrôle. Les blocs de ports sont alloués à un nouveau noeud uniquement lorsqu'il existe des blocs de ports libres. Les blocs de ports libres signifient qu'aucune connexion n'est desservie par un port mappé dans le bloc de ports.

En outre, lors de la réunion, chaque noeud recalcule le nombre de blocs qu'il peut posséder. Si un noeud contient plus de blocs qu'il n'est censé en contenir, il libère ces blocs de ports supplémentaires au noeud de contrôle au fur et à mesure de leur disponibilité. Le noeud de contrôle les attribue ensuite au noeud de données nouvellement joint.

Q. Est-il pris en charge uniquement les protocoles TCP et UDP ou SCTP ?

R. SCTP n'a jamais été pris en charge avec la PAT dynamique. Pour le trafic SCTP, la recommandation est d'utiliser une NAT d'objet réseau statique uniquement.

Q. Si un noeud est à court de ports de bloc, abandonne-t-il les paquets et n'utilise-t-il pas le prochain bloc IP disponible ?

R. Non, il ne baisse pas immédiatement. Il utilise les blocs de ports disponibles de l'adresse IP PAT suivante. Si tous les blocs de ports sur toutes les adresses IP PAT sont épuisés, le trafic est abandonné.

Q. Pour éviter la surcharge du noeud de contrôle dans une fenêtre de mise à niveau de cluster, est-il préférable de sélectionner un nouveau contrôle manuellement plus tôt (par exemple, à mi-chemin d'une mise à niveau de cluster à 4 unités), plutôt que d'attendre que toutes les connexions soient traitées sur le noeud de contrôle ?

R. Le contrôle doit être mis à jour en dernier. En effet, lorsque le noeud de contrôle exécute la version la plus récente, il ne lance pas la distribution du pool à moins que tous les noeuds exécutent la version la plus récente. En outre, lorsqu'une mise à niveau est exécutée, tous les noeuds de données dotés d'une version plus récente ignorent les messages de distribution de pool provenant d'un noeud de contrôle s'il exécute une version plus ancienne.

Pour expliquer cela en détail, considérez un déploiement de cluster avec 4 noeuds A, B, C et D avec A comme contrôle. Voici les étapes de mise à niveau typiques :

1. Téléchargez une nouvelle version sur chacun des noeuds.
2. Recharger l'unité « D ». Toutes les connexions, xlates sont déplacés vers le noeud de

sauvegarde.

3. L'unité « D » apparaît et :

a. Traite la configuration PAT

b. Décompose chaque adresse IP PAT en blocs de ports

c. Tous les blocs de ports sont à l'état non attribué

d. Ignore les anciennes versions des messages PAT de cluster reçus du contrôle

e. Redirige toutes les connexions PAT vers Primary.

4. De la même manière, affichez les autres noeuds avec la nouvelle version.

5. Commande de l’unité de rechargement « A ». Comme il n'y a pas de sauvegarde pour le contrôle, toutes les connexions existantes sont abandonnées

6. Le nouveau contrôle lance la distribution des blocs de ports dans le nouveau format

7. L’unité « A » se joint et peut accepter les messages de distribution de bloc de ports et agir en conséquence

Traitement des fragments

Symptôme :

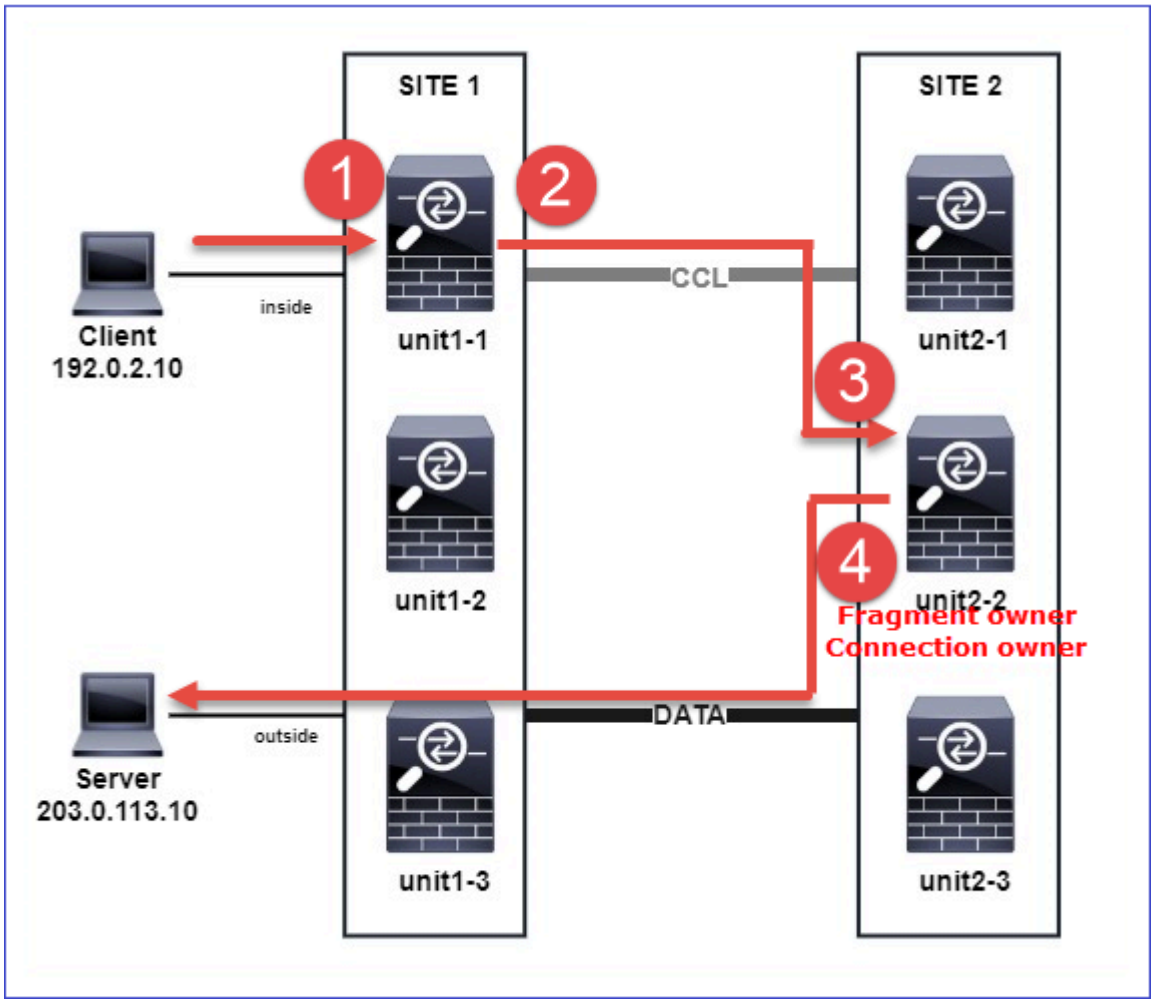
Dans les déploiements de clusters intersites, les paquets fragmentés qui doivent être traités dans un site spécifique (trafic site-local) peuvent toujours être envoyés aux unités d'autres sites, car l'un de ces sites peut avoir le propriétaire du fragment.

Dans la logique de cluster, un rôle supplémentaire est défini pour les connexions avec des paquets fragmentés : propriétaire du fragment.

Pour les paquets fragmentés, les unités de cluster qui reçoivent un fragment déterminent le propriétaire d'un fragment en fonction d'un hachage de l'adresse IP source du fragment, de l'adresse IP de destination et de l'ID de paquet. Tous les fragments sont ensuite transférés au propriétaire de fragments via la liaison de contrôle de cluster. Les fragments peuvent être équilibrés en charge vers différentes unités de cluster, car seul le premier fragment inclut le 5-tuple utilisé dans le hachage d'équilibrage de charge du commutateur. Les autres fragments ne contiennent pas les ports source et de destination et peuvent être équilibrés en charge vers d'autres unités de cluster. Le propriétaire du fragment réassemble temporairement le paquet afin qu'il puisse déterminer le directeur en fonction d'un hachage de l'adresse IP source/de destination et des ports. S'il s'agit d'une nouvelle connexion, le propriétaire du fragment devient le propriétaire de la connexion. S'il s'agit d'une connexion existante, le propriétaire du fragment transfère tous les fragments au propriétaire de la connexion via la liaison de contrôle de cluster. Le propriétaire de la connexion réassemble ensuite tous les fragments.

Considérez cette topologie avec le flux d’une requête d’écho ICMP fragmentée du client vers le

serveur :



Afin de comprendre l'ordre des opérations, il existe des captures de paquets à l'échelle du cluster sur les interfaces de liaison interne, externe et de contrôle de cluster configurées avec l'option trace. En outre, une capture de paquets avec l'option reinject-hide est configurée sur l'interface interne.

<#root>

```
firepower#
cluster exec capture capi interface inside trace match icmp any any

firepower#
cluster exec capture capir interface inside reinject-hide trace match icmp any any

firepower#
cluster exec capture capo interface outside trace match icmp any any

firepower#
cluster exec capture capccl interface cluster trace match icmp any any
```

Ordre des opérations au sein du cluster :

1. unit-1-1 dans le site 1 reçoit les paquets de requête d'écho ICMP fragmentés.

<#root>

```
firepower#
cluster exec show cap capir


unit-1-1(LOCAL)
:******************************************************
2 packets captured


1: 20:13:58.227801 802.1Q vlan#10 P0 192.0.2.10 > 203.0.113.10 icmp: echo request



2: 20:13:58.227832 802.1Q vlan#10 P0

2 packets shown
```

2. unit-1-1 sélectionne unit-2-2 dans le site 2 comme propriétaire du fragment et lui envoie des paquets fragmentés.
L'adresse MAC de destination des paquets envoyés de l'unité 1-1 à l'unité 2-2 est l'adresse MAC de la liaison CCL de l'unité 2-2.

<#root>

```
firepower#
show cap capccl packet-number 1 detail


7 packets captured
1: 20:13:58.227817
0015.c500.018f 0015.c500.029f
 0x0800 Length: 1509

192.0.2.10 > 203.0.113.10
 icmp: echo request (wrong icmp csum) (frag 46772:1475@0+) (ttl 3)
1 packet shown
firepower#
```

```
show cap capccl packet-number 2 detail


7 packets captured

2: 20:13:58.227832

0015.c500.018f 0015.c500.029f

 0x0800 Length: 637


192.0.2.10 > 203.0.113.10

 (

frag 46772

:603@1480) (ttl 3)
1 packet shown


firepower#

cluster exec show interface po48 | i MAC


unit-1-1(LOCAL):*******************************************************
MAC address 0015.c500.018f, MTU 1500
unit-1-2:*******************************************************
MAC address 0015.c500.019f, MTU 1500


unit-2-2

:*************************************************************


MAC address 0015.c500.029f, MTU 1500


unit-1-3:*******************************************************
MAC address 0015.c500.016f, MTU 1500
unit-2-1:*******************************************************
MAC address 0015.c500.028f, MTU 1500
unit-2-3:*******************************************************
MAC address 0015.c500.026f, MTU 1500
```

3. unit-2-2 reçoit, réassemble les paquets fragmentés et devient le propriétaire du flux.

```
<#root>

firepower#

cluster exec unit unit-2-2 show capture capccl packet-number 1 trace


11 packets captured

1: 20:13:58.231845 192.0.2.10 > 203.0.113.10 icmp: echo request
```

Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'inside'
Flow type: NO FLOW

I (2) received a FWD_FRAG_TO_FRAG_OWNER from (0).

Phase: 2
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'inside'

Flow type: NO FLOW

I (2) have reassembled a packet and am processing it.

Phase: 3
Type: CAPTURE
Subtype:
Result: ALLOW
Config:
Additional Information:
MAC Access list

Phase: 4
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:
MAC Access list

Phase: 5
Type: ROUTE-LOOKUP
Subtype: No ECMP load balancing
Result: ALLOW
Config:
Additional Information:
Destination is locally connected. No ECMP load balancing.
Found next-hop 203.0.113.10 using egress ifc outside(vrfid:0)

Phase: 6
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:

Input interface: 'inside'

Flow type: NO FLOW

I (2) am becoming owner

Phase: 7
Type: ACCESS-LIST
Subtype: log
Result: ALLOW
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced trust ip any any rule-id 268435460 event-log flow-end
access-list CSM_FW_ACL_ remark rule-id 268435460: PREFILTER POLICY: igasimov_prefilter1
access-list CSM_FW_ACL_ remark rule-id 268435460: RULE: r1
Additional Information:

...

Phase: 19
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 1719, packet dispatched to next module

...

Result:
input-interface: cluster(vrfid:0)
input-status: up
input-line-status: up
output-interface: outside(vrfid:0)
output-status: up
output-line-status: up

Action: allow

1 packet shown
firepower#

cluster exec unit unit-2-2 show capture capccl packet-number 2 trace

11 packets captured

2: 20:13:58.231875
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:

```
Additional Information:
Input interface: 'inside'


Flow type: NO FLOW



I (2) received a FWD_FRAG_TO_FRAG_OWNER from (0).



Result:
input-interface: cluster(vrfid:0)
input-status: up
input-line-status: up
Action: allow

1 packet shown
```

4. unit-2-2 autorise les paquets en fonction de la stratégie de sécurité et les envoie, via l'interface externe, du site 2 au site 1.

<#root>

```
firepower#

cluster exec unit unit-2-2 show cap capo



2 packets captured



1: 20:13:58.232058 802.1Q vlan#20 P0 192.0.2.10 > 203.0.113.10 icmp: echo request



2: 20:13:58.232058 802.1Q vlan#20 P0
```

Observations/mises en garde :

- Contrairement au rôle directeur, le propriétaire du fragment ne peut pas être localisé dans un site particulier. Le propriétaire du fragment est déterminé par l'unité qui reçoit à l'origine les paquets fragmentés d'une nouvelle connexion et peut être localisé sur n'importe quel site.
- Étant donné qu'un propriétaire de fragment peut également devenir le propriétaire de la connexion, afin de transférer les paquets à l'hôte de destination, il doit être en mesure de résoudre l'interface de sortie et de trouver les adresses IP et MAC de l'hôte de destination ou du saut suivant. Cela suppose que le ou les tronçons suivants doivent également être accessibles à l'hôte de destination.
- Pour réassembler les paquets fragmentés, l'ASA/FTD gère un module de réassemblage de

fragments IP pour chaque interface nommée. Pour afficher les données opérationnelles du module de réassemblage de fragments IP, utilisez la commande show fragment : <#root>

```
Interface: inside
Configuration:

Size: 200

, Chain: 24, Timeout: 5, Reassembly: virtual
Run-time stats: Queue: 0, Full assembly: 0
Drops: Size overflow: 0, Timeout: 0,
Chain overflow: 0, Fragment queue threshold exceeded: 0,
Small fragments: 0, Invalid IP len: 0,
Reassembly overlap: 0, Fraghead alloc failed: 0,
SGT mismatch: 0, Block alloc failed: 0,
Invalid IPV6 header: 0, Passenger flow assembly failed: 0
```

Dans les déploiements en cluster, le propriétaire du fragment ou le propriétaire de la connexion place les paquets fragmentés dans la file d'attente de fragments. La taille de la file d'attente de fragments est limitée par la valeur du compteur Taille (par défaut 200) qui est configuré avec la commande fragment size <size> <nameif>. Lorsque la taille de la file d'attente de fragments atteint les 2/3 de la taille, le seuil de la file d'attente de fragments est considéré comme dépassé et tous les nouveaux fragments qui ne font pas partie de la chaîne de fragments actuelle sont abandonnés. Dans ce cas, le seuil de file d'attente de fragment dépassé est incrémenté et le message syslog FTD-3-209006 est généré.

<#root>

```
firepower#

show fragment inside


Interface: inside

    Configuration:

Size: 200

, Chain: 24, Timeout: 5, Reassembly: virtual
    Run-time stats:

Queue: 133

, Full assembly: 0
    Drops: Size overflow: 0, Timeout: 8178,
           Chain overflow: 0,

Fragment queue threshold exceeded: 40802

,
           Small fragments: 0, Invalid IP len: 0,
           Reassembly overlap: 9673, Fraghead alloc failed: 0,
           SGT mismatch: 0, Block alloc failed: 0,
           Invalid IPV6 header: 0, Passenger flow assembly failed: 0
```

```
%FTD-3-209006: Fragment queue threshold exceeded, dropped TCP fragment from 192.0.2.10/21456 to 203.0.11
```

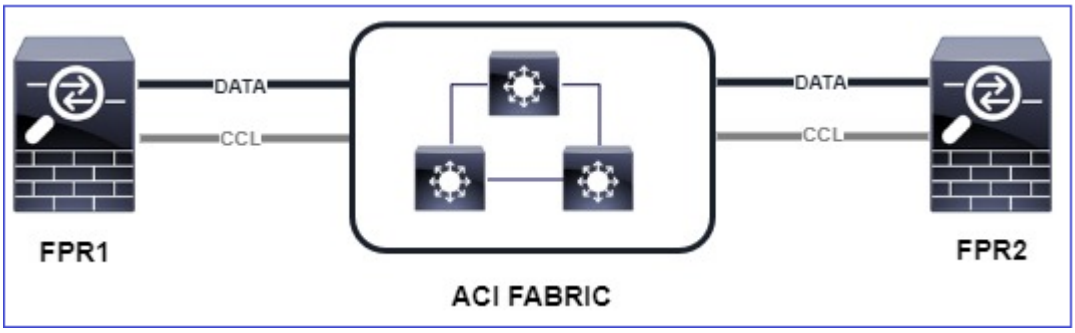
Pour contourner ce problème, augmentez la taille dans Firepower Management Center > Devices > Device Management > [Edit Device] > Interfaces > [Interface] > Advanced > Security Configuration > Override Default Fragment Setting, enregistrez la configuration et déployez les stratégies. Surveillez ensuite le compteur de file d'attente dans la sortie de la commande show fragment et l'occurrence du message syslog FTD-3-209006.

Problèmes ACI

Problèmes de connectivité intermittents dans le cluster en raison de la vérification active de la somme de contrôle de couche 4 dans le Pod ACI

Symptôme :

- Problèmes de connectivité intermittents via le cluster ASA/FTD déployé dans un Pod ACI.
- S'il n'y a qu'une seule unité dans le cluster, les problèmes de connectivité ne sont pas observés.
- Les paquets envoyés d'une unité de cluster à une ou plusieurs autres unités du cluster ne sont pas visibles dans le FXOS et les captures du plan de données des unités cibles.



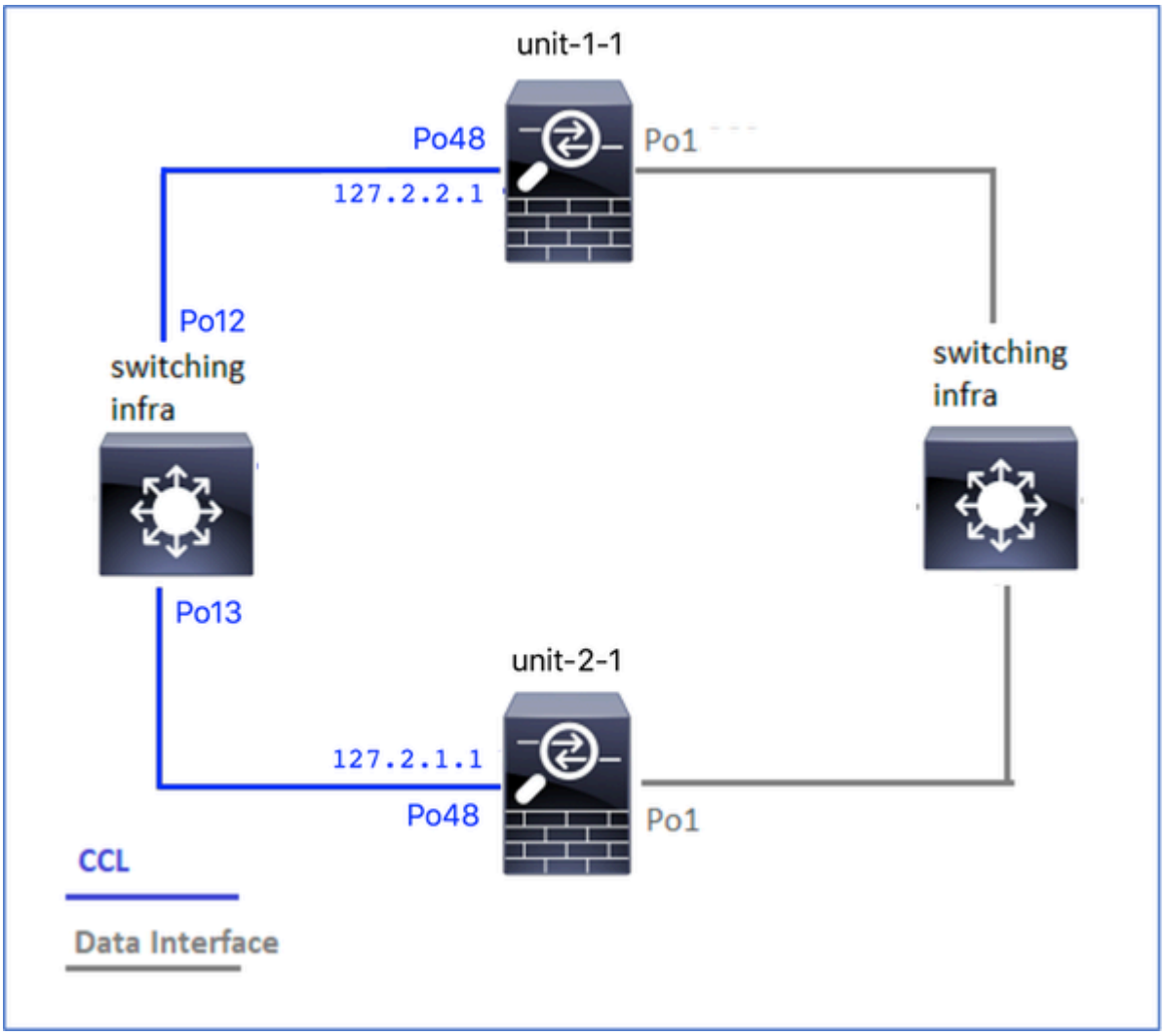
Atténuation

- Le trafic redirigé sur la liaison de contrôle de cluster n'a pas une somme de contrôle L4 correcte et ce comportement est attendu. Les commutateurs sur le chemin de liaison de contrôle de cluster ne doivent pas vérifier la somme de contrôle de couche 4. Les commutateurs qui vérifient la somme de contrôle de couche 4 peuvent entraîner l'abandon du trafic. Vérifiez la configuration du commutateur de fabric ACI et assurez-vous qu'aucune somme de contrôle de couche 4 n'est effectuée sur les paquets reçus ou envoyés via la liaison de contrôle de cluster.

Problèmes de plan de contrôle de cluster

L'unité ne peut pas joindre le cluster

Taille MTU sur CCL



Symptômes :

L'unité ne peut pas joindre le cluster et ce message s'affiche.

```
The SECONDARY has left the cluster because application configuration sync is timed out on this unit. Di
Cluster disable is performing cleanup..done.
Unit unit-2-1 is quitting due to system failure for 1 time(s) (last failure is SECONDARY application co
All data interfaces have been shutdown due to clustering being disabled. To recover either enable clust
```

Vérification/atténuation :

- Utilisez la commande show interface sur le FTD, pour vérifier que le MTU sur l'interface de liaison de contrôle de cluster est supérieur d'au moins 100 octets au MTU de l'interface de données :

<#root>

```
firepower#
```

show interface

```
Interface

Port-channel1

 "

Inside

", is up, line protocol is up
  Hardware is EtherSVI, BW 40000 Mbps, DLY 10 usec
    MAC address 3890.a5f1.aa5e,

MTU 9084


Interface

 Port-channel48

 "

cluster

", is up, line protocol is up
  Hardware is EtherSVI, BW 40000 Mbps, DLY 10 usec
    Description: Clustering Interface
    MAC address 0015.c500.028f,

MTU 9184


    IP address 127.2.2.1, subnet mask 255.255.0.
```

- Exécutez une commande ping sur la CCL, avec l'option size, pour vérifier si la configuration sur la MTU de la CCL est correcte sur tous les périphériques du chemin.

<#root>

```
firepower#

ping 127.2.1.1 size 9184
```

- Utilisez la commande show interface sur le commutateur pour vérifier la configuration MTU.

<#root>

```
Switch#

show interface



port-channel12

 is up
admin state is up,
  Hardware: Port-Channel, address: 7069.5a3a.7976 (bia 7069.5a3a.7976)
```

```
MTU 9084

bytes, BW 40000000 Kbit , DLY 10 usec


port-channel13

 is up
admin state is up,
  Hardware: Port-Channel, address: 7069.5a3a.7967 (bia 7069.5a3a.7967)


MTU 9084

bytes, BW 40000000 Kbit , DLY 10 use
```

Non-Concordance D'Interface Entre Les Unités De Cluster

Symptômes :

L'unité ne peut pas joindre le cluster et ce message s'affiche.

```
Interface mismatch between cluster primary and joining unit unit-2-1. unit-2-1 aborting cluster join.
Cluster disable is performing cleanup..done.
Unit unit-2-1 is quitting due to system failure for 1 time(s) (last failure is Internal clustering erro
All data interfaces have been shutdown due to clustering being disabled. To recover either enable clust
```

Vérification/atténuation :

Connectez-vous à l'interface utilisateur graphique de FCM sur chaque châssis, accédez à l'onglet Interfaces et vérifiez si tous les membres du cluster ont la même configuration d'interface.

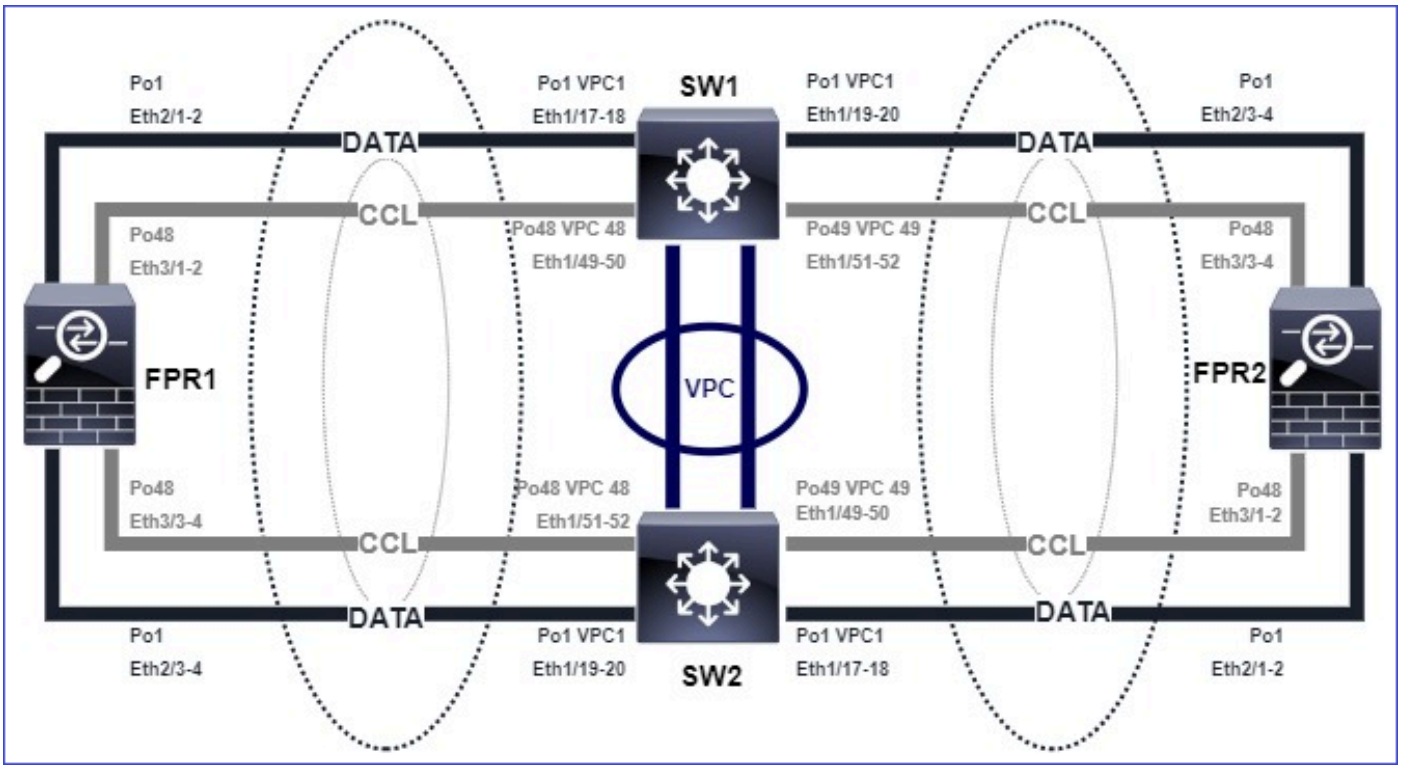
- Interfaces affectées au périphérique logique
- Vitesse d’administration des interfaces
- Duplex admin des interfaces
- État de l'interface

Problème d'interface Data/Port-Channel

Split-brain dû à des problèmes d'accessibilité sur la CCL

Symptôme :

Il y a plusieurs unités de contrôle dans le cluster. Considérez cette topologie .



Châssis 1 :

<#root>

firepower# show cluster info

```
Cluster ftd_cluster1: On
Interface mode: spanned
```

This is "unit-1-1" in state PRIMARY

```
ID : 0
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TU5H
CCL IP : 127.2.1.1
CCL MAC : 0015.c500.018f
Last join : 07:30:25 UTC Dec 14 2020
Last leave: N/A
Other members in the cluster:
Unit "unit-1-2" in state SECONDARY
ID : 1
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TU4D
CCL IP : 127.2.1.2
CCL MAC : 0015.c500.019f
Last join : 07:30:26 UTC Dec 14 2020
Last leave: N/A
Unit "unit-1-3" in state SECONDARY
ID : 3
Site ID : 1
Version : 9.15(1)
```

```
Serial No.: FLM2102THJT
CCL IP : 127.2.1.3
CCL MAC : 0015.c500.016f
Last join : 07:31:49 UTC Dec 14 2020
Last leave: N/A
```

Châssis 2 :

<#root>

```
firepower# show cluster info
```

```
Cluster ftd_cluster1: On
Interface mode: spanned
```

```
This is "unit-2-1" in state PRIMARY
```

```
ID : 4
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TUN1
CCL IP : 127.2.2.1
CCL MAC : 0015.c500.028f
Last join : 11:21:56 UTC Dec 23 2020
Last leave: 11:18:51 UTC Dec 23 2020
Other members in the cluster:
Unit "unit-2-2" in state SECONDARY
ID : 2
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2102THR9
CCL IP : 127.2.2.2
CCL MAC : 0015.c500.029f
Last join : 11:18:58 UTC Dec 23 2020
Last leave: 22:28:01 UTC Dec 22 2020
Unit "unit-2-3" in state SECONDARY
ID : 5
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TUML
CCL IP : 127.2.2.3
CCL MAC : 0015.c500.026f
Last join : 11:20:26 UTC Dec 23 2020
Last leave: 22:28:00 UTC Dec 22 2020
```

Vérification :

- Utilisez la commande ping pour vérifier la connectivité entre les adresses IP de liaison de contrôle de cluster (CCL) des unités de contrôle.

<#root>

```
firepower# ping 127.2.1.1


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 127.2.1.1, timeout is 2 seconds:

?????
Success rate is 0 percent (0/5)
```

- Vérifiez la table ARP.

<#root>

```
firepower# show arp


cluster 127.2.2.3 0015.c500.026f 1
cluster 127.2.2.2 0015.c500.029f 1
```

- Dans les unités de contrôle, configurez et vérifiez les captures sur les interfaces CCL.

<#root>

```
firepower# capture capccl interface cluster
firepower# show capture capccl | i 127.2.1.1


2: 12:10:57.652310 arp who-has 127.2.1.1 tell 127.2.2.1
41: 12:11:02.652859 arp who-has 127.2.1.1 tell 127.2.2.1
74: 12:11:07.653439 arp who-has 127.2.1.1 tell 127.2.2.1
97: 12:11:12.654018 arp who-has 127.2.1.1 tell 127.2.2.1
126: 12:11:17.654568 arp who-has 127.2.1.1 tell 127.2.2.1
151: 12:11:22.655148 arp who-has 127.2.1.1 tell 127.2.2.1
174: 12:11:27.655697 arp who-has 127.2.1.1 tell 127.2.2.1
```

Atténuation :

- Assurez-vous que les interfaces port-channel CCL sont connectées à des interfaces port-channel distinctes sur le commutateur.
- Lorsque des vPC (Virtual Port-Channels) sont utilisés sur des commutateurs Nexus, assurez-vous que les interfaces CCL Port-Channel sont connectées à des vPC différents et que la configuration vPC n'a pas d'état de cohérence défaillant.
- Assurez-vous que les interfaces port-channel CCL se trouvent dans le même domaine de diffusion et que le VLAN CCL est créé et autorisé sur les interfaces.

Voici un exemple de configuration de commutateur :

<#root>

```
Nexus#
show run int po48-49


interface port-channel48
description FPR1

switchport access vlan 48



vpc 48


interface port-channel49
description FPR2

switchport access vlan 48



vpc 49



Nexus#
 show vlan id 48


VLAN Name Status Ports
---- ----------- --------- -------------------------------

48   CCL active Po48, Po49, Po100, Eth1/53, Eth1/54


VLAN Type Vlan-mode
---- ----- ----------
48 enet CE

1  Po1  up success success 10,20

48 Po48 up success success 48



49 Po49 up success success 48



<#root>

Nexus1#
```

```
show vpc brief


Legend:
(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id : 1
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status : success


Per-vlan consistency status : success



Type-2 consistency status : success


vPC role : primary
Number of vPCs configured : 3
Peer Gateway : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled
Auto-recovery status : Disabled
Delay-restore status : Timer is off.(timeout = 30s)
Delay-restore SVI status : Timer is off.(timeout = 10s)

vPC Peer-link status
---------------------------------------------------------------------
id Port Status Active vlans
-- ---- ------ --------------------------------------------------
1 Po100 up 1,10,20,48-49,148

vPC status
----------------------------------------------------------------------------
id Port Status Consistency Reason Active vlans
-- ---- ------ ----------- ------ ------------
1 Po1 up success success 10,20


48 Po48 up success success 48



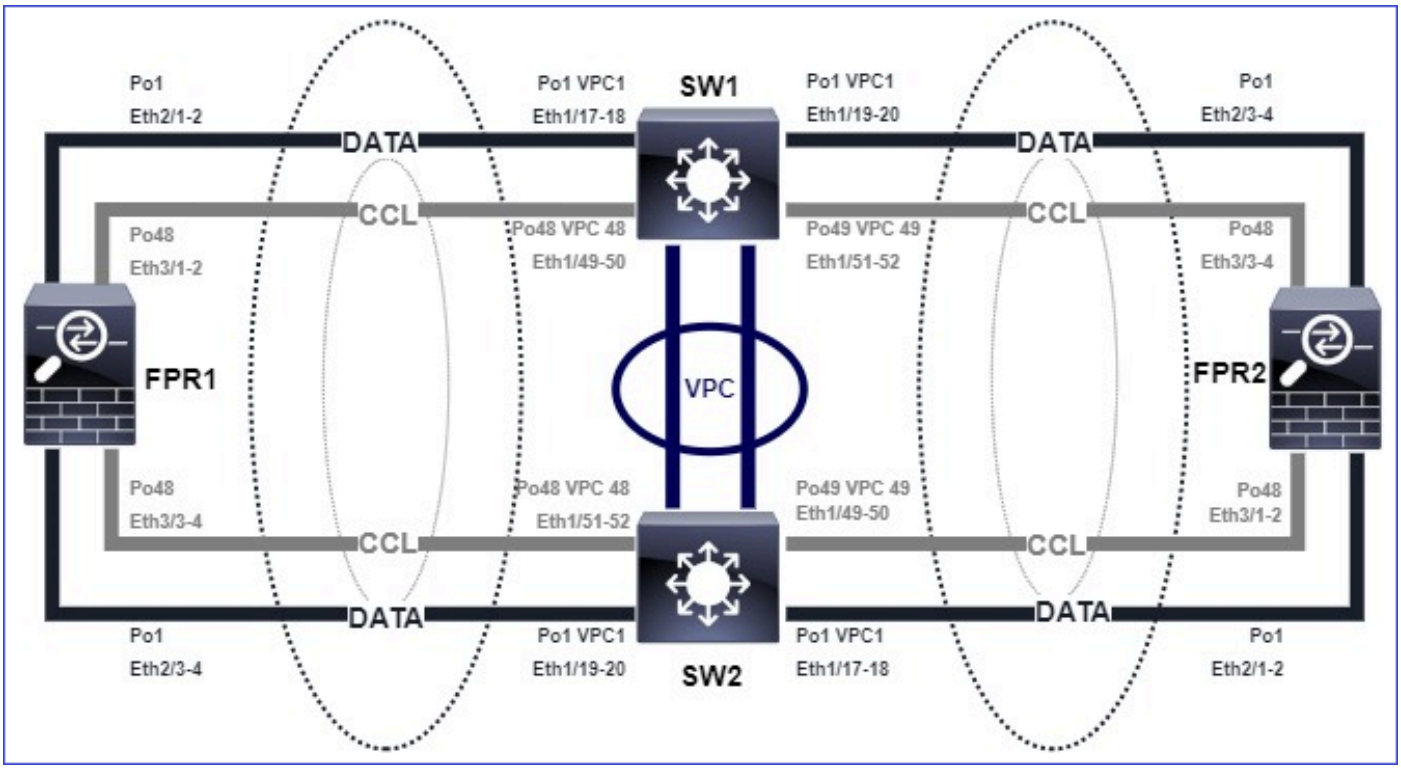
49 Po49 up success success 48
```

Cluster désactivé en raison d'interfaces Port Channel de données suspendues

Symptôme :

Une ou plusieurs interfaces port-canal de données sont suspendues. Lorsqu'une interface de données activée par l'administrateur est suspendue, toutes les unités de cluster du même châssis sont retirées du cluster en raison d'un échec du contrôle d'intégrité de l'interface.

Considérez cette topologie :



Vérification :

- Vérifiez la console de l'unité de contrôle.

<#root>

```
firepower#
Beginning configuration replication to
```

SECONDARY unit-2-2

```
End Configuration Replication to SECONDARY.
Asking SECONDARY unit
```

unit-2-2

```
 to quit because it
```

failed interface health

```
 check 4 times (last failure on
```

Port-channel1

```
). Clustering must be manually enabled on the unit to rejoin.
```

- Vérifiez le résultat des commandes show cluster history et show cluster info trace module hc dans la ou les unités concernées.

<#root>

```
firepower# Unit is kicked out from cluster because of interface health check failure.
```

```
Cluster disable is performing cleanup..done.
All data interfaces have been shutdown due to clustering being disabled. To recover either enable clust

Cluster unit unit-2-1 transitioned from SECONDARY to DISABLED


firepower#
```

show cluster history

```
==========================================================================
From State To State Reason
==========================================================================

12:59:37 UTC Dec 23 2020
ONCALL SECONDARY_COLD Received cluster control message

12:59:37 UTC Dec 23 2020
SECONDARY_COLD SECONDARY_APP_SYNC Client progression done

13:00:23 UTC Dec 23 2020
SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application configuration sync done

13:00:35 UTC Dec 23 2020
SECONDARY_CONFIG SECONDARY_FILESYS Configuration replication finished

13:00:36 UTC Dec 23 2020
SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done
```

13:01:35 UTC Dec 23 2020

SECONDARY_BULK_SYNC DISABLED Received control message DISABLE (interface health check failure)

<#root>

```
firepower#
```

show cluster info trace module hc

```
Dec 23 13:01:36.636 [INFO]cluster_fsm_clear_np_flows: The clustering re-enable timer is started to expi
Dec 23 13:01:32.115 [INFO]cluster_fsm_disable: The clustering re-enable timer is stopped.
```

Dec 23 13:01:32.115 [INFO]Interface Port-channel1 is down

- Vérifiez le résultat de la commande show port-channel summary dans l'interpréteur de commandes fxos.

<#root>

```
FPR2(fxos)#
```

```
 show port-channel summary


Flags: D - Down P - Up in port-channel (members)
I - Individual H - Hot-standby (LACP only)
s - Suspended r - Module-removed
S - Switched R - Routed
U - Up (port-channel)
M - Not in use. Min-links not met
------------------------------------------------------------------------------
Group Port-Channel Type Protocol Member Ports

------------------------------------------------------------------------------


1 Po1(SD) Eth LACP Eth2/1(s) Eth2/2(s) Eth2/3(s) Eth2/4(s)


48 Po48(SU) Eth LACP Eth3/1(P) Eth3/2(P) Eth3/3(P) Eth3/4(P)
```

Atténuation :

- Assurez-vous que tous les châssis ont le même nom de groupe de cluster et le même mot de passe.
- Assurez-vous que les interfaces port-channel disposent d'interfaces membres physiques administrativement activées avec la même configuration duplex/vitesse dans tous les châssis et commutateurs.
- Dans les clusters intra-site, assurez-vous que la même interface port-channel de données dans tous les châssis est connectée à la même interface port-channel sur le commutateur.
- Lorsque des canaux de port virtuels (vPC) sont utilisés dans les commutateurs Nexus, assurez-vous que la configuration vPC n'a pas d'état de cohérence défaillant.
- Dans les clusters intra-site, assurez-vous que la même interface port-canal de données dans tous les châssis est connectée au même vPC.

Problèmes de stabilité des clusters

Suivi FXOS

Symptôme :

L'unité quitte le cluster.

Vérification/atténuation :

- Utilisez la commande show cluster history pour voir quand l'unité a quitté le cluster.

<#root>

```
firepower#

show cluster history
```

- Utilisez ces commandes pour vérifier si le FXOS dispose d'un traceback.

```
<#root>

FPR4150#

connect local-mgmt


FPR4150 (local-mgmt)#

dir cores
```

- Collectez le fichier de base généré au moment où l'unité a quitté le cluster et fournissez-le au TAC.

Disque plein

Si l'utilisation du disque dans la partition /ngfw d'une unité de cluster atteint 94 %, l'unité quitte le cluster. La vérification de l'utilisation du disque a lieu toutes les 3 secondes :

```
<#root>

> show disk


Filesystem Size Used Avail Use% Mounted on
rootfs 81G 421M 80G 1% /
devtmpfs 81G 1.9G 79G 3% /dev
tmpfs 94G 1.8M 94G 1% /run
tmpfs 94G 2.2M 94G 1% /var/volatile
/dev/sda1 1.5G 156M 1.4G 11% /mnt/boot
/dev/sda2 978M 28M 900M 3% /opt/cisco/config
/dev/sda3 4.6G 88M 4.2G 3% /opt/cisco/platform/logs
/dev/sda5 50G 52M 47G 1% /var/data/cores
/dev/sda6 191G 191G 13M

100% /ngfw


cgroup_root 94G 0 94G 0% /dev/cgroups
```

Dans ce cas, le résultat de la commande show cluster history affiche :

```
<#root>

15:36:10 UTC May 19 2021
PRIMARY Event: Primary unit unit-1-1 is quitting
                    due to

diskstatus

 Application health check failure, and
```

```
                    primary's application state is down
```

ou

```
14:07:26 CEST May 18 2021
SECONDARY DISABLED Received control message DISABLE (application health check failure)
```

Vous pouvez également vérifier l'échec de la manière suivante :

<#root>

```
firepower#
```

show cluster info health

```
Member ID to name mapping:
0 - unit-1-1(myself) 1 - unit-2-1

               0   1
Port-channel48 up up
Ethernet1/1 up up
Port-channel12 up up
Port-channel13 up up

Unit overall            healthy healthy
Service health status:
                        0       1
```

diskstatus (monitor on) down   down

```
snort (monitor on)      up      up
Cluster overall         healthy
```

En outre, si le disque est ~100 %, l'unité peut avoir des difficultés à se joindre au cluster jusqu'à ce qu'un espace disque soit libéré.

Protection Contre Les Débordements

Toutes les 5 minutes, chaque unité de cluster vérifie l'utilisation du processeur et de la mémoire par l'unité locale et l'unité homologue. Si l'utilisation est supérieure aux seuils du système (CPU LINA 50 % ou mémoire LINA 59 %), un message d'information s'affiche dans :

- Syslogs (FTD-6-748008)
- Fichier log/cluster_trace.log, par exemple :

```
<#root>

firepower#

more log/cluster_trace.log | i CPU


May 20 16:18:06.614 [INFO][

CPU load 87%

 | memory load 37%] of module 1 in chassis 1 (unit-1-1) exceeds overflow protection threshold [

CPU 50% | Memory 59%

]. System may be oversubscribed on member failure.
May 20 16:18:06.614 [INFO][CPU load 87% | memory load 37%] of chassis 1 exceeds overflow protection thr
May 20 16:23:06.644 [INFO][CPU load 84% | memory load 35%] of module 1 in chassis 1 (unit-1-1) exceeds
```

Le message indique qu'en cas de panne d'une unité, les autres ressources de l'unité peuvent être sursouscrites.

Mode simplifié

Comportement sur les versions FMC antérieures à la version 6.3 :

- Vous enregistrez chaque noeud de cluster individuellement sur FMC.
- Vous formez ensuite un cluster logique dans FMC.
- Pour chaque ajout de noeud de cluster, vous devez enregistrer manuellement le noeud.

Post-6.3 FMC :

- La fonction de mode simplifié vous permet d'enregistrer l'ensemble de la grappe sur FMC en une seule étape (il suffit d'enregistrer n'importe quel noeud de la grappe).

Gestionnaire minimal pris en charge	Périphériques gérés	Version minimale du périphérique géré prise en charge requise	Remarques
FMC 6.3	Clusters FTD sur FP9300 et FP4100 uniquement	6.2.0	Il s'agit d'une fonctionnalité FMC uniquement

Avertissement : Une fois que le cluster est formé sur FTD, vous devez attendre l'enregistrement automatique pour démarrer. Vous ne devez pas essayer d'enregistrer les noeuds de cluster manuellement (Ajouter un périphérique), mais utiliser l'option Rapprocher.

Symptôme :

Échecs d'enregistrement des noeuds

- Si l'enregistrement du noeud de contrôle échoue pour une raison quelconque, le cluster est supprimé de FMC.

Atténuation :

Si l'enregistrement du noeud de données échoue pour une raison quelconque, il existe 2 options :

1. À chaque déploiement sur le cluster, FMC vérifie si des noeuds de cluster doivent être enregistrés, puis lance l'enregistrement automatique de ces noeuds.
2. Une option Reconcile est disponible sous l'onglet de résumé du cluster (Périphériques > Gestion des périphériques > onglet Cluster > lien View Cluster Status). Une fois l'action de réconciliation déclenchée, FMC démarre l'enregistrement automatique des noeuds qui doivent être enregistrés.

Informations connexes

- Mise en grappe pour Firepower Threat Defense
- Cluster ASA pour châssis Firepower 4100/9300
- À propos du clustering sur le châssis Firepower 4100/9300
- Mise en grappe de pare-feu de nouvelle génération Firepower - BRKSEC-3032
- Analysez les captures de pare-feu Firepower pour résoudre efficacement les problèmes de réseau

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.